ITEM NO. 3

To recommend Curriculum, Model Study Plan, and Course Syllabi, for BS Data Science in the light of the Undergraduate Policy 2023 by Higher Education Commission (HEC) of Pakistan at University of the Punjab, Lahore, Punjab University Gujranwala campus, Punjab University Jhelum campus, Punjab University Pothohar campus, and the affiliated colleges.

Updated Curriculum
of
Bachelor of Science

in Data Science BS (DS)

By

Department of Data Science,
Faculty of Computing and Information Technology

University of the Punjab,
Lahore, Pakistan

BS in Data Science

Program Title:	BS Data Science
Department:	Department of Data Science (DDS)
Faculty:	Faculty of Computing and Information Technology

Admission Eligibility Criteria for BS (DS)

Same as already approved by Academic Council

Degree Completion Requirement for BS Data Science

- At minimum 130 credit hours are required for award of BS degree in Data Science.
- A minimum 2.0 CGPA on a scale of 4.0 is required for the award of BS Data Science Degree
- Maximum duration allowed by the HEC/PU guidelines
- Non-credit Internship (as per HEC/PU guidelines)
- Practical Learning Lab (as per HEC/PU guidelines)
- Non-credit course/s (if any)

Two years Associate Degree in Data Science

- The students after successful completion of 04 semesters in BS Data Science Program may exit with Associate Degree in Data Science. However, the completion of all requirements for the award of Associate Degree in Data Science (Credit Hours, CGPA, and Compulsory Courses) needs to be fulfilled.

Areas Covered in the BS Data Science

Numeric Code (N)	**Group Code**	**Couse Group**	**HEC**	**PU**
0	MD	Math Deficiency*		0 (2)
1	CC	Computing Core	46 (14)	46 (19)
2	DD	Data Science Core	18 (6)	18 (6)
3,4	ED	Data Science Elective	21 (7)	21 (7)
5	MS	Mathematics & Supporting Courses	12 (4)	12 (4)
6,9	GE	General Education Requirement	30 (12)	30 (12)
7	UE	Elective Supporting Courses	3 (1)	3 (1)
8	HQ	Quran Translation	0	4 (8)
		Total	**130 (44)**	**134 (57)**

*: non credited deficiency course

Course Coding Scheme

Code-YNS

Code = MD, CC, DS, ES, MS, GE, UE, HQ

Y = Year of Offering = 1, 2, 3, 4

N = Numeric Code = 0, 1, …, 9

S = Serial Number (Resets with Year) = 0, 1, 2, …,9

Lab Codes are same as related *theory course code* with '-L' as the suffix

Computing Core (CC) Courses

Total Credit hours: 46 (19)

Sr. No.	Course Title	Cr Hrs	Lab
1	Programming Fundamentals	3	0
2	Programming Fundamentals Lab	0	1
3	Object Oriented Programming	3	0
4	Object Oriented Programming Lab	0	1
5	Database Systems	3	0
6	Database Systems Lab	0	1
7	Digital Logic Design	2	0
8	Digital Logic Design Lab	0	1
9	Data Structures	3	0
10	Data Structures Lab	0	1
11	Information Security	2	1
12	Artificial Intelligence	2	1
13	Computer Networks	2	1
14	Software Engineering	3	0
15	Computer Organization and Assembly Language	2	1
16	Operating Systems	2	1
17	Analysis of Algorithms	3	0
18	Final Year Project-1	0	2
19	Final Year Project-2	0	4
	Total	**30**	**16**

Domain Core (DD) Courses

Total Credit hours: 18 (6)

Sr. No.	Course Title	Cr Hrs	Lab
1	Introduction to Data Science	2	1
2	Advanced Statistics	2	1
3	Data Mining and Machine Learning	2	1
4	Data Visualization	2	1
5	Data Warehousing and Business Intelligence	2	1
6	Parallel and Distributed Computing	2	1
	Total	**12**	6

Mathematics and Supporting (MS) Courses

Total Credit hours: 12 (4)

Sr. No.	Course Title	Cr Hrs	Lab
1	Multivariable Calculus	3	0
2	Linear Algebra	3	0
3	Probability and Statistics	3	0
4	Technical and Business Writing	3	0
	Total	**12**	**0**

General Education Requirement (GE) Courses

Total Credit hours: 30 (12)

Sr. No.	Course Title	Sub – Category	Cr Hrs	Lab
1	Applications of Information and Communication Technologies		2	1
2	Functional English		3	0
3	Expository Writing		3	0
4	Discrete Structures	Quantitative Reasoning – I	3	0
5	Calculus and Analytic Geometry	Quantitative Reasoning – II	3	0
6	Islamic Studies*		2	0
7	Ideology and Constitution of Pakistan		2	0
8	Introduction to Management	Social Science	2	0
9	Applied Physics	Natural Science	2	1
10	Professional Practices		2	0
11	Civic and Community Engagement	Arts and Humanities	2	0
12	Entrepreneurship		2	0
	Total		**28**	**2**

*: Alternative course for Islamic Studies will be offered to the non-Muslim students.

Elective Supporting (ES) Courses

Total Credit hours: 3 (1) from the list

Sr. No.	Course Title	Cr Hrs	Lab
1	Introduction to Marketing	3	0
	Total	**3**	**0**

Note: New courses may be added to the list with the approval of the Convener, Board of Studies, Department of Data Science, University of the Punjab, Lahore.

Quran Translation (QT) Courses

Total Credit hours: 4(8)

Sr. No.	Course Title	Cr Hrs	Lab
1	Quran Translation I		0.5
2	Quran Translation II		0.5
3	Quran Translation III		0.5
4	Quran Translation IV		0.5
5	Quran Translation V		0.5
6	Quran Translation VI		0.5
7	Quran Translation VII		0.5
8	Quran Translation VIII		0.5
	Total		**4**

Note: Alternative courses will be offered through the university to non-Muslim students.

Domain Elective (DE) Courses

Total Credit hours: 21 (7)

Sr. No.	Course Title	Cr Hrs	Lab
1	Web Technologies	2	1
2	Deep Learning	2	1
3	Natural Language Processing	2	1
4	Computer Vision	2	1
5	Big Data Analytics	2	1
6	Theory of Automata and Formal Languages	2	1
7	Software Construction and Development	2	1
	Total	**14**	**7**

Note: New courses may be added to the list with the approval of the Convener, Board of Studies, Department of Data Science, University of the Punjab, Lahore.

.

MATHEMATICS DEFICIENCY (MD) Courses

Total Credit hours: 0 (2)

Sr. No.	Course Title	Cr Hrs	Lab
1	Math Deficiency - I	0	
2	Math Deficiency - II	0	

Note: These are PASS/FAIL based courses with 3 hours teaching per week.

Model Scheme of Study for BS Data Science

Semester I					
Sr#	**Code**	**Course Title**	**Type**	**Prerequisite**	**Credit hours**
1	HQ-001	Quran Translation I	HQ		0.5
2	GE-190	Functional English	GE		3 (3-0)
3	GE-168	Ideology and Constitution of Pakistan	GE		2 (2-0)
4	GE-160	Applications of Information and Communication Technologies	GE		3 (2-1)
5	GE-163	Islamic Studies	GE		2 (2-0)
6	CC-112	**Programming Fundamentals**	CC		3 (3-0)
7	*CC-112-L*	***Programming Fundamentals Lab***	CC		1 (0-1)
8	*MD-001*	Math Deficiency – I*	MD		0
		** (Taught 3 hours per week)*			
				Total	**14.5**

Semester II					
Sr#	**Code**	**Course Title**	**Type**	**Prerequisite**	**Credit hours**
1	HQ-002	Quran Translation II	HQ		0.5
2	GE-169	Applied Physics	GE		3 (2-1)
3	MS-251	Probability and Statistics	MS		3 (3-0)
4	GE-199	Expository Writing	GE		3 (3-0)
5	CC-110	**Digital Logic Design**	CC		2 (2-0)
6	*CC-110-L*	***Digital Logic Design Lab***	CC		1 (0-1)
7	CC-211	**Object Oriented Programming**	CC	Programming Fundamentals	3 (3-0)
8	*CC-211-L*	***Object Oriented Programming Lab***	CC	Programming Fundamentals	1 (0-1)
9	*MD-002*	Math Deficiency – II*	MD		0
		** (Taught 3 hours per week)*			
				Total	**16.5**

Semester III					
Sr#	**Code**	**Course Title**	**Type**	**Prerequisite**	**Credit hours**
1	HQ-003	Quran Translation III	HQ		0.5
2	GE-162	Calculus and Analytic Geometry	GE		3 (3-0)
3	CC-210	Computer Organization and Assembly Language	CC	Digital Logic Design	3 (2-1)
4	GE-167	Discrete Structures	GE		3 (3-0)
5	CC-213	**Data Structures**	CC	Object-Oriented Programming	3 (3-0)
6	*CC-213-L*	***Data Structures Lab***	CC	Object-Oriented Programming	1 (0-3)
7	DD-221	Introduction to Data Science	DD	Programming Fundamentals	3 (2-1)
8	MS-252	Linear Algebra	MS		3 (3-0)
				Total	**19.5**

Semester IV

Sr#	Code	Course Title	Type	Prerequisite	Credit hours
1	HQ-004	Quran Translation IV	HQ		0.5
2	MS-253	Multivariable Calculus	MS	Calculus and Analytical Geometry	3 (3-0)
3	DD-222	Data Visualization	DD		3 (2-1)
4	CC-215	**Database Systems**	CC		3 (3-0)
5	*CC-215-L*	***Database Systems Lab***	CC		1 (0-3)
6	CC-310	Artificial Intelligence	CC	Object Oriented Programming	3 (2-1)
7	CC-214	Computer Networks	CC		3 (2-1)
8	DD-223	Advanced Statistics	DD	Probability & Statistics	3 (2-1)
				Total	**19.5**

Semester V

Sr#	Code	Course Title	Type	Prerequisite	Credit hours
1	HQ-005	Quran Translation V	HQ		0.5
2	CC-311	Operating Systems	CC	Data Structures	3 (2-1)
3	GE-363	Civics and Community Engagement	GE		2 (2-0)
4	CC-313	Analysis of Algorithms	CC	Data Structures	3 (3-0)
5	DD-321	Data Warehousing and Business Intelligence	DD	Database Systems	3 (2-1)
6	ED-323	Web Technologies	ED	Object Oriented Programming	3 (2-1)
7	DD-322	Data Mining and Machine Learning	DD	Introduction to Data Science	3 (3-0)
				Total	**17.5**

Semester VI

Sr#	Code	Course Title	Type	Prerequisite	Credit hours
1	HQ-006	Quran Translation VI	HQ		0.5
2	ES-492	Introduction to Marketing	UE		3 (3-0)
3	CC-308	Information Security	CC		3 (2-1)
4	CC-303	Software Engineering	CC		3 (3-0)
5	ED-321	Deep Learning	ED		3 (2-1)
7	ED-322	Natural Language Processing	ED		3 (2-1)
8	ED-335	Computer Vision	ED		3 (2-1)
				Total	**18.5**

Semester VII

Sr#	Code	Course Title	Type	Prerequisite	Credit hours
1	HQ-007	Quran Translation VII	HQ		0.5
2	MS-254	Technical and Business Writing	MS		3 (3-0)
3	GE-362	Entrepreneurship	GE		2 (2-0)
4	ED-333	Theory of Automata and Formal Languages	ED		3 (2-1)
5	ED-421	Big Data Analytics	ED		3 (2-1)
6	ED-324	Software Construction and Development	ED	Object Oriented Programming	3 (2-1)
7	CC-401	Final Year Project-1	CC		2 (0-6)

				Total	**16.5**
Semester VIII					
Sr#	**Code**	**Course Title**	**Type**	**Prerequisite**	**Credit hours**
1	HQ-008	Quran Translation VIII	HQ		0.5
2	ES-492	Introduction to Management	GE		2 (2-0)
3	DD-409	Parallel and Distributed Computing	DD	Operating Systems	3 (2-1)
4	GE-402	Professional Practices	GE		2 (2-0)
5	CC-402	Final Year Project-2	CC	FYP-1	4 (0-12)
				Total	**11.5**

Adoption of OBEs for BS Data Science Program

Outcome-based education is about preparing students for life, not simply getting them ready for college or employment.

Department Vision

To become a premier in institute of Data Science that is dedicated to excellence in teaching, research, and fostering entrepreneurship.

Department Mission

The mission of the DDS is to effectively prepare our students in Data Science and related fields, so that they can serve community with skill, knowledge and high character, and be a source of pride for the Institute and to the homeland (Pakistan).

1. Graduate Attributes (GAs)

Graduate attributes are statements that describe the set of skills, knowledge, and attitude that the Department of Data Science expects from its graduates. The department will monitor its performance from the data gathered while evaluating the attainment of GAs by its graduates. By virtue of this data the institute will be able to assess the quality of its graduates and take steps to carry out necessary improvements in weak areas that are affecting better attainment of GAs.
GAs defined by NCEAC and adopted by the Department of Data Science at University of the Punjab are in alignment with the Graduate Attributes laid down in the Seoul Accord document D.5 for computing professionals. The range qualifier in several attribute statements uses the notion of complex computing problem, or the notion of complex activity.

- **GA-1 Academic Education:** Completion of an accredited program of study designed to prepare graduates as computing professionals.
- **GA-2 Knowledge for Solving Computing Problems:** Apply knowledge of computing fundamentals, knowledge of a computing specialization, and mathematics, science, and domain knowledge appropriate for the computing specialization to the abstraction and conceptualization of computing models from defined problems and requirements.
- **GA-3 Problem Analysis:** Identify and solve *complex* computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.

- **GA-4 Design/Development of Solutions:** Design and evaluate solutions for *complex* computing problems, and design and evaluate systems, components, or processes that meet specified needs.
- **GA-5 Modern Tool Usage:** Create, select, or adapt and then apply appropriate techniques, resources, and modern computing tools to *complex* computing activities, with an understanding of the limitations.
- **GA-6 Individual and Teamwork:** Function effectively as an individual and as a member or leader of a team in multidisciplinary settings.
- **GA-7 Communication:** Communicate effectively with the computing community about *complex* computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions.
- **GA-8 Computing Professionalism and Society:** Understand and assess societal, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practice
- **GA-9 Ethics:** Understand and commit to professional ethics, responsibilities, and norms of professional computing practice.
- **GA-10 Life-long Learning:** Recognize the need, and have the ability, to engage in independent learning for continual development as a computing professional.

2. Program Educational Objectives (PEOs)

Program Educational Objectives	
PEO No	**Description**
PEO-1	Solve real-world problems by applying mathematical and computational approaches.
PEO-2	Understand, design, and develop computer-based solutions and data components to fulfill the customer needs.
PEO-3	Present, analyze, and identify patterns in large-sized datasets (numeric, text, image, video, audio)
PEO-4	Understand professional, ethical, and social responsibilities and effective communication
PEO-5	Harness the benefits of lifelong useful traits like working in groups, leading and managing projects and building capacity to learn new tools and techniques as per need.
PEO-6	Make best use of essential entrepreneurial skills that result in producing next generation of industry leaders.

3. Program Learning Outcomes (PLOs)

The Department of Data Science ensures that the graduates are equipped with the right set of attributes that will make them desirable to potential employers. Problem Solving, Critical Thinking, Creativity, Communication, Teamwork, Adaptability, and Life-long Learning are all considered as crucial attributes for employability. These graduate attributes (GAs) are in line with student outcomes defined by Seoul Accord; therefore, the department of computer science at CUI has adapted SOs from Seoul Accord which is listed in Table below:

PLO #	Program Learning Outcomes (PLOs)	Description
PLO-1	Academic Education	To prepare graduates as computing professionals
PLO-2	Knowledge for Solving Computing Problems	Apply knowledge of computing fundamentals, knowledge of a computing specialization, and mathematics, science, and domain knowledge appropriate for the computing specialization to the abstraction and conceptualization of computing models from defined problems and requirements.
PLO-3	Problem Analysis	Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.
PLO-4	Design/ Development of Solutions	Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.
PLO-5	Modern Tool Usage	Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.
PLO-6	Individual and Teamwork	Function effectively as an individual and as a member or leader in diverse teams and in multi-disciplinary settings.
PLO-7	Communication	Communicate effectively with the computing community and with society at large about complex computing activities by being able to comprehend and write effective

		reports, design documentation, make effective presentations, and give and understand clear instructions.
PLO-8	Computing Professionalism and Society	Understand and assess societal, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practice
PLO-9	Ethics	Understand and commit to professional ethics, responsibilities, and norms of professional computing practice.
PLO-10	Life-long Learning	Recognize the need, and have the ability, to engage in independent learning for continual development as a computing professional

4. Mapping between PEOs and PLOs

The following is the mapping between PLOs and PEOs.

PLOs to PEOs Mapping							
PLO	**Title**	**PEOs**					
		PEO-1	**PEO-2**	**PEO-3**	**PEO-4**	**PEO-5**	**PEO-6**
PLO-1	Academic Education	✓					
PLO-2	Knowledge for Solving Computing Problems	✓		✓			
PLO-3	Problem Analysis	✓	✓	✓			
PLO-4	Design/ Development of Solutions	✓	✓	✓			
PLO-5	Modern Tool Usage		✓	✓			
PLO-6	Individual and Teamwork				✓	✓	
PLO-7	Communication			✓	✓	✓	
PLO-8	Computing Professionalism and Society			✓	✓		
PLO-9	Ethics				✓		✓
PLO-10	Life-long Learning					✓	✓

5. Mapping between Courses and PEOs

A mapping between courses offered in the BS Data Science program and PLOs is defined.

			PEOs					
S#	**CCode**	**Course Titles**	**1**	**2**	**3**	**4**	**5**	**6**
1	GE-190	Functional English				✓	✓	
2	GE-168	Ideology and Constitution of Pakistan				✓	✓	
3	GE-160	Applications of ICT	✓					
4	GE-163	Islamic Studies/University Equivalent				✓	✓	
5	CC-112	Programming Fundamentals	✓	✓				
6	GE-169	Applied Physics	✓	✓				
7	MS-251	Probability and Statistics	✓		✓			
8	GE-199	Expository Writing				✓	✓	
9	CC-110	Digital Logic Design	✓	✓				
10	CC-211	Object Oriented Programming	✓	✓				
11	GE-162	Calculus & Analytical Geometry	✓		✓			
12	CC-210	Computer Organization & Assembly	✓					
13	GE-167	Discrete Structures	✓					
14	CC-213	Data Structures	✓	✓	✓			
15	DD-221	Introduction to Data Science	✓	✓				
16	MS-252	Linear Algebra	✓		✓			
17	MS-253	Multivariate Calculus	✓		✓			
18	DD-222	Data Visualization & Graphics		✓	✓			
19	CC-215	Database Systems		✓				
20	CC-310	Artificial Intelligence		✓	✓			
21	CC-214	Computer Networks		✓				
22	DD-223	Advanced Statistics	✓					
23	CC-311	Operating Systems		✓				
24	GE-363	Civics & Community Engagement				✓	✓	✓
25	CC-313	Analysis of Algorithms		✓	✓			
26	DD-321	Data Warehousing & BI		✓	✓			
27	DD-322	Data Mining and Machine Learning		✓	✓			
28	DS-3WW	Data Science Elective I	✓	✓	✓			
29	SS-3XX	Social Science Elective				✓	✓	✓
30	CC-308	Information Security	✓	✓	✓			
31	DS-303	Software Engineering		✓	✓			
32	DS-4YZ	Data Science Elective II	✓	✓	✓			
33	DS-3ZZ	Data Science Elective III	✓	✓	✓			
34	DS-3WX	Data Science Elective IV	✓	✓	✓			
35	MS-254	Technical and Business Writing				✓		✓
36	GE-362	Entrepreneurship				✓	✓	✓
37	DS-402	Data Science Elective V	✓	✓	✓			
38	CC-401	Final Year Project-1	✓	✓	✓	✓	✓	✓
39	DS-4XY	Data Science Elective VI	✓	✓	✓			
40	DS-3YY	Data Science Elective VII	✓	✓	✓			
41	SS-4XX	Social Science Elective (SS)				✓	✓	
42	DD-409	Parallel and Distributed Computing	✓	✓	✓			
43	GE-402	Professional Practices				✓		
44	CC-402	Final Year Project-2	✓	✓	✓	✓	✓	✓

6. Mapping between Courses and PLOs

A mapping between courses offered in the BS Data Science program and PLOs is defined.

			PLOs									
S#	**CCode**	**Course Titles**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	GE-190	Functional English			✓			✓	✓			
2	GE-168	Ideology and Constitution of Pakistan							✓	✓		✓
3	GE-160	Applications of ICT	✓	✓								
4	GE-163	Islamic Studies/University Equivalent						✓	✓	✓	✓	✓
5	CC-112	Programming Fundamentals	✓	✓								
6	GE-169	Applied Physics	✓	✓								
7	MS-251	Probability and Statistics	✓	✓	✓							
8	GE-199	Expository Writing			✓			✓	✓	✓		
9	CC-110	Digital Logic Design	✓	✓								
10	CC-211	Object Oriented Programming	✓	✓	✓	✓						✓
11	GE-162	Calculus & Analytical Geometry	✓	✓	✓	✓						
12	CC-210	Computer Organization & Assembly Language	✓	✓	✓	✓						
13	GE-167	Discrete Structures	✓	✓								
14	CC-213	Data Structures	✓	✓	✓	✓						
15	DD-221	Introduction to Data Science	✓	✓	✓		✓					
16	MS-252	Linear Algebra	✓	✓	✓	✓						
17	MS-253	Multivariate Calculus	✓	✓	✓	✓						
18	DD-222	Data Visualization & Graphics	✓	✓	✓	✓	✓					
19	CC-215	Database Systems	✓	✓	✓	✓	✓					
20	CC-310	Artificial Intelligence	✓	✓	✓	✓	✓					
21	CC-214	Computer Networks	✓	✓	✓							
22	DD-223	Advanced Statistics	✓	✓	✓	✓						
23	CC-311	Operating Systems	✓	✓	✓	✓	✓					
24	GE-363	Civics & Community Engagement				✓		✓	✓	✓	✓	
25	CC-313	Analysis of Algorithms	✓	✓	✓	✓						
26	DD-321	Data Warehousing & BI	✓	✓	✓	✓	✓					
27	DD-322	Data Mining and Machine Learning	✓	✓	✓	✓	✓					
28	DS-3WW	Data Science Elective I	✓	✓	✓	✓	✓					✓
29	SS-3XX	Social Science Elective	✓	✓	✓	✓		✓	✓	✓	✓	
30	CC-308	Information Security	✓	✓	✓	✓	✓			✓		
31	DS-303	Software Engineering	✓	✓	✓	✓				✓		
32	DS-4YZ	Data Science Elective II	✓	✓	✓	✓	✓					✓
33	DS-3ZZ	Data Science Elective III	✓	✓	✓	✓	✓					✓
34	DS-3WX	Data Science Elective IV	✓	✓	✓	✓	✓					✓
35	MS-254	Technical and Business Writing				✓		✓	✓	✓	✓	✓
36	GE-362	Entrepreneurship				✓		✓	✓	✓	✓	✓
37	DS-402	Data Science Elective V	✓	✓	✓	✓	✓					✓
38	CC-401	Final Year Project-1	✓	✓	✓	✓						
39	DS-4XY	Data Science Elective VI	✓	✓	✓	✓						✓
40	DS-3YY	Data Science Elective VII	✓	✓	✓	✓						✓
41	SS-4XX	Social Science Elective (SS)				✓		✓	✓	✓	✓	
42	DD-409	Parallel and Distributed Computing	✓	✓	✓	✓						
43	GE-402	Professional Practices				✓		✓	✓	✓	✓	✓
44	CC-402	Final Year Project-2	✓	✓	✓	✓						

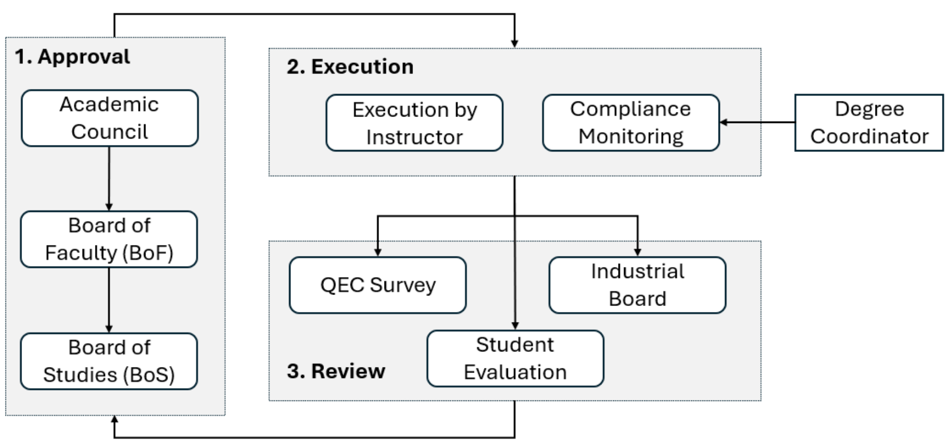
7. BLOOM's Taxonomy

In the BS Data Science curricula, the Cognitive domain of Bloom's Taxonomy is used to assess student mastery of learning outcomes. The Department of Data Science is using the following six levels for the Cognitive domain, progressing from the lowest order processes to the highest.

- **Remembering** - Retrieving, recalling, or recognizing information from memory. Students can recall or remember information. Note: This process is the most basic thinking skill.
- **Understanding** - Constructing meaning or explaining material from written, spoken, or graphic sources. Students can explain ideas or concepts.
- **Applying** - Using learned materials or implementing materials in new situations. Students can use/apply information in a new way.
- **Analyzing** - Breaking material or concepts into parts, determining how the parts relate or interrelate to one another or to an overall structure or purpose. Students can distinguish between different parts.
- **Evaluating** - Assessing, making judgments and drawing conclusions from ideas, information, or data. Students can justify a stand or decision.
- **Creating** - Putting elements together or reorganizing them into a new way, form or product. Students can create a new product. Note: This process is the most difficult mental function.

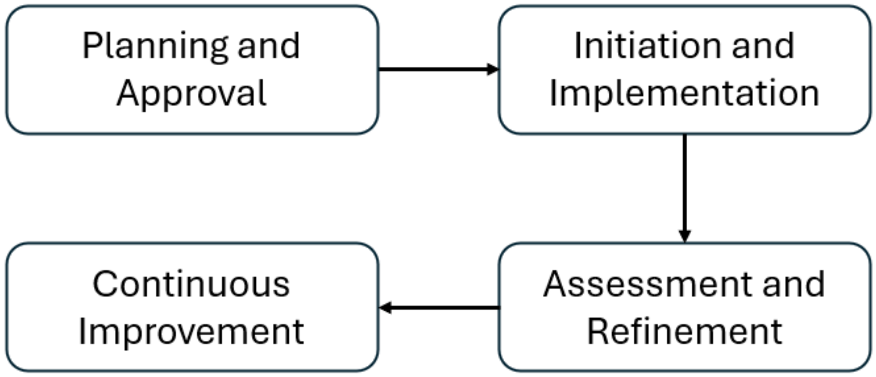
8. PLOs and CLOs Assessment Roadmap Framework

The framework proposed for the assessment of PLOs and CLOs is the following. Note that the QEC survey will be held two years after the graduation of the first batch. The assessment results and the findings of the survey will be presented to the industrial board for deliberations.



9. Roadmap for Adopting OBE for BSDS

The four-phased proposed roadmap for adopting OBE based system for BS in Data Science.



- **Planning and approval:**
    - Identify and clearly define the desired learning outcomes for PLOs, and CLO.
    - Align the curriculum with the defined outcomes. Ensure that each course contributes to the overall program outcomes.
    - Develop instructional materials and teaching methods that align with the desired outcomes. Incorporate active learning, collaborative activities, and real-world problem-solving tasks.
    - Approval of programs from relevant bodies.

- **Initiation and Implementation**
    - *Awareness.* Introduce students to the OBE framework, explaining the focus on outcomes and the importance of their active participation in the learning process.
    - *Faculty training.* Conduct training sessions and workshops for faculty to familiarize them with OBE principles and teaching strategies.

- **Assessment and Refinement**
    - Implement continuous assessment methods to monitor student progress and provide timely feedback. This can include quizzes, assignments, projects, and peer assessments.
    - Conduct end-of-course assessments to evaluate whether the learning outcomes have been achieved. This can include exams, capstone projects, and practical assessments.
    - Collect data on student performance and analyze it to assess the effectiveness of the teaching methods and the achievement of learning outcomes.
    - Establish a feedback mechanism to gather input from students, faculty, and other stakeholders on the effectiveness of the OBE system.

- **Continuous Improvements**
    - Regularly review the curriculum, teaching methods, and assessment strategies to identify areas for improvement.

- Engage stakeholders in the continuous improvement process to ensure that the program remains relevant and effective.
- Provide ongoing professional development opportunities for faculty to stay updated with the latest teaching strategies and technological advancements.
- Based on the feedback and data analysis, make necessary adjustments to the curriculum, teaching methods, and assessments to enhance the overall effectiveness of the OBE system.

Detailed Course Outlines for BS in DATA SCIENCE

<table>
<tr><td>Program</td><td colspan="3">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="3">GE-160</td></tr>
<tr><td>Course Title</td><td colspan="3">Applications of Information and Communication Technologies</td></tr>
<tr><td rowspan="2">Credit Hours</td><td colspan="2">Theory</td><td>Lab</td></tr>
<tr><td colspan="2">2</td><td>1</td></tr>
<tr><td>Lecture Duration</td><td colspan="3">60 minutes (1.0 Hours) 2 lectures per week, 3 hours lab session per week</td></tr>
<tr><td>Semester</td><td colspan="3">1</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td colspan="2">Courses</td><td>Knowledge</td></tr>
<tr><td colspan="2">Nil</td><td>Nil</td></tr>
<tr><td>Follow Up Courses</td><td colspan="3"></td></tr>
<tr><td colspan="4">Course Learning Outcomes (CLOs)</td></tr>
<tr><td>CLO No</td><td colspan="2">Course Learning Outcome</td><td>Bloom Taxonomy</td></tr>
<tr><td>CLO-1</td><td colspan="2">Understand basics of computing technology</td><td>C1 (Knowledge)</td></tr>
<tr><td>CLO-2</td><td colspan="2">Do number systems conversions and arithmetic</td><td>C2 (Understand)</td></tr>
<tr><td>CLO-3</td><td colspan="2">Have knowledge of types of software</td><td>C2 (Understand)</td></tr>
<tr><td>CLO-4</td><td colspan="2">Have knowledge of computing related technologies</td><td>C3 (Apply)</td></tr>
<tr><td>Objectives</td><td colspan="3">1. This is an introductory course in Computer Science designed for beginners.
2. Apart from leading the participants through a whirlwind history of computing, the course also develops a feel for web programming through a series of lectures that help the students develop their own web page.</td></tr>
</table>

	3. Main objective of the course is to build an appreciation for the fundamental concepts in computing and to become familiar with popular PC productivity software.
Learning Outcomes	• Understand basics of computing technology • Do number systems conversions and arithmetic • Have knowledge of types of software • Have knowledge of computing related technologies
Contents	1. Brief History of Computer 1.1. Four Stages of History 2. Computer Elements and Software Types 2.1. Processor, Memory, Hardware, Software 2.2. Application Software its uses and Limitations 2.3. System Software its Importance and its Types 3. Types of Computer 3.1. Super Compute 3.2. Mainframe Compute 3.3. Mini Compute 3.4. Micro Compute 4. Organizing Computer Facility 4.1. Centralized Computing Facility 4.2. Distributed Computing Facility 4.3. Decentralized Computing Facility 5. Input Devices 5.1. Keyboard and its Types, 5.2. Terminal (Dump, Smart, Intelligent), 5.3. Dedicated Data Entry 5.4. Pointing Devices, Voice Input, 6. Output Devices

	6.1. Soft- Hard 6.2. Copies, Monitors and its Types, Printers and its Types, Plotters, 6.3. Computer Virus and its Forms, 6.4. Storage Units, 6.5. Primary and Secondary Memories, 7. RAM and its Types 7.1. Popular types of RAM 8. Cache Memory 8.1. Cache Memory Importance 8.2. Type of Cache Memory 9. Hard Disks, Working of Hard Disk 9.1. Diskettes, RAID, 9.2. Optical Disk Storages (DVD, CD ROM), 9.3. Magnetic Types, Backup System, 10. Data Communications 10.1. Data Communication Model 10.2. Data Transmission 10.3. Digital and Analog Transmission 10.4. Modems 10.5. Asynchronous and Synchronous Transmission 10.6. Simplex. Half Duplex, Full Duplex Transmissions 11. Communications 11.1. Medias (Cables, Wireless) 11.2. Protocols, Network Topologies (Star, Bus, Ring) 11.3. LAN, WAN, and MAN 12. Internet 12.1. A Brief History 12.2. Birthplace of ARPA Net 12.3. Web Browser

<table>
<tr><td></td><td colspan="4">12.4. Internet Services provider
12.5. Function and Features of Browse
12.6. Search Engines</td></tr>
<tr><td>Teaching-learning Strategies</td><td colspan="4">• Interactive class session
• Hands on practices in class
• Brainstorming and group discussion sessions</td></tr>
<tr><td>Assignments</td><td colspan="4">• Paper based written assignments 4
• Coding HTML and CSS 3</td></tr>
<tr><td rowspan="4">Assessment and Examinations</td><td>Sr. #</td><td>Elements</td><td>Weightage</td><td>Details</td></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.</td></tr>
<tr><td>Textbooks</td><td colspan="4">• Sinha, P. K., & Sinha, P. (2010). Computer fundamentals. BPB publications.</td></tr>
</table>

	• Morley, D., & Parker, C. S. (2014). Understanding computers: Today and tomorrow, comprehensive. Cengage Learning.
Reference Material/Suggested Readings	• Livesley, R. K. (2017). An introduction to automatic digital computers. Cambridge University Press. • Zawacki-Richter, O., & Latchem, C. (2018). Exploring four decades of research in Computers & Education. Computers & Education, 122, 136-152.
Notes	• Academic integrity is expected of all students. Plagiarism or cheating in any assessment will result in at least an F grade in the course, and possibly more severe penalties. • You bear all the responsibility for protecting your assignments from plagiarism. If anyone else submits your assignment or uses your code in his/her assignment, you will be considered equally responsible. • The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester. • There is no makeup for a missed sessional grading instruments like quizzes, assignments, and homework's.

Detailed Lecture wise plan

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
1	1	Brief History of Computer Four Stages of History	Ch#1	
	2	Computer Elements and Software Types Processor, Memory, Hardware, Software	Ch#1	Assignment
2	3	Computer Elements and Software Types Processor, Memory, Hardware, Software	Ch#2	
	4	System Software its Importance and its Types, Types of Computer, Super Computer, Mainframe Compute, Mini Compute, Micro Compute	Ch#3	Quiz
3	5	Organizing Computer Facility	Ch#4	
	6	Centralized Computing	Ch#4	
4	7	Distributed Computing	Ch#4	Assignment
	8	Input Devices Keyboard and its Types, Terminal (Dump, Smart, Intelligent), Dedicated Data Entry Pointing Devices, Voice Input,	Ch#5	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
5	9	Output Devices Soft- Hard Copies, Monitors and its Types, Printers and its Types, Plotters, Computer Virus and its Forms,	Ch#5	Quiz
	10	Storage Units, Primary and Secondary Memories, RAM and its Types Popular types of RAM	Ch#5	
6	11	Cache Memory Cache Memory Importance	Ch#5	
	12	Type of Cache Memory Hard Disks, Working of Hard Disk	Ch#5	
7	13	Diskettes, RAID, Optical Disk Storages (DVD, CD ROM), Magnetic Types, Backup System,	Ch#5	
	14	Data Communications	Ch#6	Assignment
8	15	Data Communication Model	Ch#6	
	16	Data Transmission	Ch#6	
Midterm Exams				
9	17	Digital and Analog Transmission	Ch#9	Quiz
	18	Modems		

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
10	19	Asynchronous and Synchronous Transmission	Ch#9	
	20	Simplex. Half Duplex, Full Duplex Transmissions	Ch#9	
11	21	Communications	Ch#9	
	22	Protocols, Network Topologies (Star, Bus, Ring)	Ch#11	Assignment
12	23	LAN, WAN, and MAN	Ch#11	
	24	Internet	Ch#12	
13	25	A Brief History	Ch#12	
	26	Birthplace of ARPA Net	Ch#12	
14	27	Web Browser	Ch#13	
	28	Internet Services provider	Ch#14	Quiz
15	29	Internet Services provider	Ch#14	
	30	Function and Features of Browse	Ch#14	
16	31	Search Engines	Ch#14	
	32	Advanced Search Engines		
Final Exam				

Program	BS Data Science	
Course Code	GE-167	
Course Title	Discrete Structures	
Credit Hours	**Theory**	**Lab**
	3	0
Lecture Duration	90 minutes (1.5 Hours), 2 lectures per week	
Semester	1	
Pre-requisites	**Courses**	**Knowledge**
	Nil	Python lab Setup
Follow Up Courses	Nil	
Course Learning Outcomes (CLOs)		
CLO No	**Course Learning Outcome**	**Bloom Taxonomy**
CLO-1	Understand the key concepts of Discrete Structures such as Sets, Permutations, Relations, Graphs and Trees etc.	C2 (Understand)
CLO-2	Apply formal logic proofs and/or informal, but rigorous, logical reasoning to real problems, such as predicting the behavior of software or solving problems such as puzzles.	C3 (Apply)
CLO-3	Apply discrete structures into other computing problems such as formal specification, verification, databases, artificial intelligence, and cryptography.	C3 (Apply)
CLO-4	Differentiate various discrete structures and their relevance within the context of computer science, in the areas of data structures and algorithms, in particular	C4 (Differentiate)
Aims and Objectives	1. This course aims to equip the students with an understanding and appreciation of the discrete mathematical structures that	

	appear in most computer science problems as well as in other relateddisciplines 2. In particular, this course seeks to develop critical thinking skills and the skills for comprehendingmathematical arguments as well for writing rigorous proofs.
Learning Outcomes	• It seeks to polish basic problem solving skills through study of combinatorial problems, predicatecalculus and logical reasoning, abstract algebra, and graph structures.
Syllabus	Logics and Proofs, Sets, Functions, Sequences, Algorithms, Recursion, Probability, Graphs and Trees
Contents	**Chapter 01: The Foundations: Logic and Proofs** 1.1 Introduction to Discrete Structures 1.2 Application of Propositional Logic 1.3 Propositional Equivalences 1.4 Predicates and Quantifiers 1.5 Nested Quantifiers 1.6 Rules of Inference **Chapter 02: Basic Structures: Sets, Functions, Sequences, Sums, and Matrices** 2.1 Sequences and Summations 2.2 Cardinality of Sets 2.3 Sets, Functions **Chapter 03: Algorithms** 3.1 Algorithms 3.2 The Growth of Functions 3.3 Complexity of Algorithms **Chapter 05: Induction and Recursion**

	5.1 Mathematical Induction 5.2 Recursive Algorithms **Chapter 06: Counting** 6.1 The Basic of Counting 6.2 The Pigeonhole Principle 6.3 Permutations and Combinations 6.4 Binomial Coefficients and Identities **Chapter 07: Discrete Probability** 7.1 An Introduction to Discrete Probability 7.2 Probability Theory **Chapter 09: Relations** 9.1 Relations and their properties 9.2 Closure of Relations 9.3 Equivalence Relation 9.4 Partial Ordering **Chapter 10: Graphs** 10.1 Graphs and Graph Models 10.2 Graph Isomorphism, Graph Connectivity 10.3 Eulerian, Hamiltonian paths and circuits 10.4 Shortest path problems (dijkstra algorithm) **Chapter 11: Trees** 11.1 Introduction to Trees, properties of trees 11.2 Applications of Trees 11.3 Tree Traversals
Assignments	• Late submissions will not be accepted. • Assignments should be turned in at the start of the class. • Zero credit for turning in questions other than the assigned questions.

	Sr. #	Elements	Weightage	Details
Assessment and Examinations	1	Quizzes + Assignment + Term Project	25%	There will be a graded quiz and assignments. The term project will be framed so as to test the concepts involved in the lectures.
	2	Midterm Assessment	35%	It takes place at the mid-point of the semester.
	3	Final Assessment	40%	It takes place at the end of the semester.
Textbooks	• Discrete Mathematics & its Applications with Combinatory and Graph Theory, 7th edition,Kenneth H. Rosen			
Reference Material/Suggested Readings	• N/A			

Detailed Lecture wise plan

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
1	1	Introduction to discretemathematics	Ch#(01)	
	2	Introduction to propositional logic	Ch#(01)	**Assign-1**
2	3	Application of propositional logic	Ch#(01)	**Quiz#1**
	4	Consistent SystemSpecifications Propositional Equivalences	Ch#(01)	
3	5	Propositional Equivalences	Ch#(01)	**Quiz#2**
	6	Propositional Equivalences	Ch#(01)	
4	7	Predicates and Quantifiers	Ch#(01)	**Assign-2**
	8	Nested quantifiers Rules of inference	Ch#(01)	
5	9	Sets, Functions	Ch#(02)	**Quiz#3**
	10	Cardinality of Sets	Ch#(02)	
6	11	Sequences and Summations	Ch#(02)	
	12	Algorithms	Ch#(03)	
7	13	The Growth of Functions	Ch#(03)	**Quiz#4**
	14	Complexity of Algorithms	Ch#(03)	
8	15	Mathematical induction	Ch#(05)	**Assign-3**
	16	Recursion and Recursivealgorithms	Ch#(05)	
MID TERM				
9	17	The Basic of Counting	Ch#(06)	
	18	The Pigeonhole principle.	Ch#(06)	
10	19	Permutations and combinations	Ch#(06)	**Assign-4**
	20	Binomial coefficients and identities	Ch#(06)	
11	21	An Introduction to Discrete Probability	Ch#(07)	
	22	Probability Theory	Ch#(07)	**Quiz#5**

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
12	23	Relations, Closure of relations	Ch#(09)	**Assign-5**
	24	Equivalence Relation	Ch#(09)	
13	25	Partial Ordering	Ch#(09)	
	26	Graphs and Graph representation	Ch#(10)	**Quiz#7**
14	27	Graph Isomorphism, Graph Connectivity	Ch#(10)	
	28	Eulerian, Hamiltonian paths and circuits	Ch#(10)	
15	29	Shortest path problems (dijkstra algorithm)	Ch#(10)	**Quiz#8**
	30	Shortest path problems (dijkstra algorithm)	Ch#(10)	
16	31	Introduction to trees, properties of trees	Ch#(11)	**Assign-6**
	32	Applications of Trees, Tree traversals	Ch#(11)	
FINAL TERM				

<table>
<tr><td>Program</td><td colspan="3">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="3">CC-112</td></tr>
<tr><td>Course Title</td><td colspan="3">Programming Fundamentals</td></tr>
<tr><td rowspan="2">Credit Hours</td><td colspan="2">Theory</td><td>Lab</td></tr>
<tr><td colspan="2">3</td><td>1</td></tr>
<tr><td>Lecture Duration</td><td colspan="3">90 minutes (1.5 Hours), 2 lectures per week, 1 LAB per week</td></tr>
<tr><td>Semester</td><td colspan="3">1</td></tr>
<tr><td rowspan="2">Pre-requisites course / skills</td><td colspan="2">Courses</td><td>Knowledge</td></tr>
<tr><td colspan="2">Nil</td><td>Nil</td></tr>
<tr><td>Follow Up Courses</td><td colspan="3">Object Oriented Programming</td></tr>
<tr><td colspan="4">Course Learning Outcomes (CLOs)</td></tr>
<tr><td>CLO No</td><td colspan="2">Course Learning Outcome</td><td>Bloom Taxonomy</td></tr>
<tr><td>CLO-1</td><td colspan="2">Understand basic problem-solving steps and logic constructs</td><td>C2 (Understand)</td></tr>
<tr><td>CLO-2</td><td colspan="2">Apply basic programming concepts</td><td>C3 (Apply)</td></tr>
<tr><td>CLO-3</td><td colspan="2">Design and implement algorithms to solve real-world problems</td><td>C3 (Solve)</td></tr>
<tr><td>Objectives</td><td colspan="3">1. Students should be able to translate their basic pseudo-code/flow-charts into some programming language that computer can understand so that they can get real feel of their efforts.
2. Student can translate of their logic into some programming language.
3. Students can learn basic principles of attacking a problem, a bit of performance factor and some basic structured design principles.</td></tr>
</table>

	4. Students should be ready to take Object Oriented Programming course.
Learning Outcomes	• Students can write a program. • Students should be able to translate a computation problem into program. • Student can familiar with C++. • Student can design and implement algorithms to solve real world problems.
Syllabus	Topics: Flowcharts/Pseudo Codes, Basic C++ Language Constructs: Data types, Variable and Constants, Operator and Expressions, Input and Output (I/O), Formatted I/O, Escape Sequences. Structured Programming in C Language: Decision making using if control structure, Repetition using for and do while, multiple selection using switch and logical operators. Procedural Programming in C Language: functions, prototype, parameter and arguments, call by value and call by reference, library and header files, scope and life time of variables (storage classes), recursion. Composite data types arrays: definition, processing, and passing of array to a function, multidimensional arrays, searching and sorting. Pointers: pointer definition, pointer arithmetic, constant pointers, pointer and arrays. Strings: string and characters, string conversion functions, Dynamic Memory Allocation. User Defined Data Types: structures, definition, initialization, accessing members of structures, typedef, union and bitwise operators, enumerations. C File Processing: files and streams, Sequential Access File, Random Access File, Secondary Storage I/O. Miscellaneous Topics: Command Line Arguments.
Contents	1. Flow Charts/Pseudo Code 1.1. Sequence, Conditions, Repetition

	2. C++ Programming Language Introduction 3. Hello world in C++, COUT 3.1. Difference between Variables and Literals, Identifiers 4. Data Types 5. Cin, extraction operator 6. Formatted Output 7. Selection: 7.1. Relational operators and expression 7.2. If, if-else, switch 8. Repetition: 8.1. Loop, While, For, Do while 8.2. Sentinel-controlled loops, Nested loops 8.3. Increment and decrement operator 9. Function: 9.1. Defining, Calling, function prototype, passing arguments by value 9.2. Local and global variables, Static variables, 9.3. Default arguments 9.4. Overloading functions 10. Arrays: 10.1. Parallel Arrays, 2D Arrays 11. Pointers 12. CString 13. Structs, Union 14. Text and Binary File I/O
Teaching-learning Strategies	• Interactive class session • Hands on practices in class • Brainstorming and Group discussion sessions • Coding in LABS

<table>
<tr><td>Assignments</td><td colspan="4">Coding Assignments 5</td></tr>
<tr><td rowspan="4">Assessment and Examinations</td><td>Sr. #</td><td>Elements</td><td>Weightage</td><td>Details</td></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.</td></tr>
<tr><td>Textbooks</td><td colspan="4">• Gaddis, T., & Sengupta, P. (2012). Starting Out with C++: From Control Structures Through Objects. Pearson.</td></tr>
<tr><td>Reference Material/Suggested Readings</td><td colspan="4">• R1. Reference from different books enlisted in reference material will be given as required or lecture notes for reading will be provided.
• R2. Malik, D. S. (2011). JavaTM Programming: From Problem Analysis to Program Design. Cengage Learning.
• R3. Ritchie, D. M., Kernighan, B. W., & Lesk, M. E. (1988). The C programming language. Englewood Cliffs: Prentice Hall.</td></tr>
</table>

	• Handout provided by the teacher.

Detailed Lecture wise plan

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
1	1	Introduction to the Course: Role of Course in the Degree. Pseudo Code: Sequence		
	2	Flow Charts/Pseudo Code: Conditions	Conditional Structure	**Assign-1**
2	3	Flow Charts/Pseudo Code: Repetition	Repetition Structure	**Quiz#1**
	4	Flow Charts/Pseudo Code: Repetition	Repetition Structure	
3	5	Flow Charts/Pseudo Code: Repetition	Repetition Structure	**Quiz#2**
	6	C++ Programming Language: Introduction, History, and Significance. Setting up Programming/Compiling Environment Program Development Life Cycle: Editing, Compiling, Linking, Loading and Execution. Source file, Object Files, and Executable files.	A- (1.3)	**Assign-2**
4	7	Introduction to C++: Hello world program, cout, insertion operator, Escape sequences, #include; Difference between Variables and Literals, Identifiers	A- (2.1-2.3)	**Quiz#3**
	8	Data types: short, int, long, char, float, double, bool; sizeof operator; Variable assignment and initialization; Arithmetic operators; Comments;	A- (2.5-2.7, 2.9-2.16)	
5	9	Taking input: cin, extraction operator and its properties; Concept buffered input; Reading c-strings, Arithmetic expressions: operator precedence and associativity; Library functions: power; Overflow, Underflow, Type coercion, Type casting, Named constants: const, #define; Multiple assignment	A- (3.1-3.6)	**Quiz#4**

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
	10	Formatted output: setw, setprecision, fixed, showpoint, left, right Formatted input: cin.get, cin.getline, cin.ignore; Random number generation	A- (3.7-3.13)	
6	11	Selection: Relational operators and expressions, Truth values, if, if-else, nested if, if-else if, Logical operators, Input validation, Scope	A- (4.1-4.12)	
	12	Selection: Comparing strings (strcmp), Conditional operator, switch statement	A- (4.13-4.16)	**Quiz#5**
7	13	Repetition: Increment and decrement operators, while loop, Input validation	A- (5.1-5.6)	
	14	Repetition: Sentinel-controlled loops, Nested loops, break, continue	A- (5.7, 5.8, 5.11-5.13)	
8	15	Functions: Motivation, Defining, Calling, Function prototype, Passing arguments by value	A- (6.1-6.9)	**Quiz#6**
	16	Passing arguments by reference, Introduction to Pointers	R1	
9	17	Functions: Local and global variables, Static variables, Default arguments Overloading functions	A- (6.10-6.12, 6.14-6.16)	
	18	Arrays: Introduction, Declaration, Subscripts, Input and output, No bounds checking, Array initialization	A- (7.1-7.5, 7.7)	**Quiz#7**

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
10	19	Parallel Arrays Searching (Linear, Binary), Sorting (Selection Sort) Pointer Data Type Significance Pointer Arithmetic, working of index operator	A- (7.6, 9)	
	20	Two-D arrays, Mapping formulae for 1-D and 2-D array Character Arrays Array of character arrays (2-D character arrays)	A- (7.8-7.10)	**Assign-3**
11	21	Pointers: Using const with pointers, Heap: Dynamic memory allocation (new and delete), Dangling pointers	A- (9)	**Quiz#8**
	22	Pointers: Dangling pointers, Memory leak, Pointer to pointer (Multiple indirection), Dynamically allocating a 2-D array	A- (9)	
12	23	Alias C-Strings: Library functions (strlen, strcpy, strcat, strncpy, strncat, strstr), Conversion (atoi, atol, atof, itoa), strtok	A- (10)	**Assign-4**
	24	Structs: Declaration, Accessing members, Initialization, Arrays of structs, Nested structs, Passing/returning structs to/from functions	A- (11)	
13	25	Structs: Pointer to struct, Dynamic allocation, Pointer member variables;	A- (11)	**Quiz#9**
	26	Structs: Case Study	R1	
14	27	Unions, Enumerated Data Type	A- (11)	**Quiz#10**
	28	Streams: Text, Binary Text data reading writing	A- (12.1-12.4)	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
15	29	Text File	R1	**Assign-5**
	30	Binary File I/O: get, put, read, write Binary File I/O:	A- (12)	**Quiz # 11**
16	31	Creating records with structures, Random access files (seekg, seekp, tellg, tellp) Binary File I/O: Opening a file for both input and output	R1	
	32	Binary File and Structs Case Study	R1	

Program	BS Data Science	
Course Code	CC-211	
Course Title	Object Oriented Programming	
Credit Hours	**Theory**	**Lab**
	3	1
Lecture Duration	90 minutes (1.5 Hours), 2 lectures per week, 3 hours lab session per week	
Semester	2	
Pre-requisites	**Courses**	**Knowledge**
	Programming Fundamentals	Students should know how to program in C++, Structural programming in C++.
Follow Up Courses	Data Structures	
Course Learning Outcomes (CLOs)		
CLO No	**Course Learning Outcome**	**Bloom Taxonomy**
CLO-1	Understand principles of object-oriented paradigm.	C2 (Understand)
CLO-2	Identify the objects & their relationships to build object-oriented solution.	C3 (Identify)
CLO-3	Model a solution for a given problem using object-oriented principles.	C3 (Apply)
CLO-4	Examine an object-oriented solution.	C4 (Examine)
Objectives	1. To equip the learner with the philosophy and necessary skills to formulate solutions of real world problems using object oriented paradigm. 2. Justify the philosophy of object-oriented design and the concepts of encapsulation, abstraction, inheritance, and polymorphism.	

	3. Strong concepts of object manipulation and dynamic memory allocation within classes
Learning Outcomes	• Students can formulate solutions of real world problems using object oriented paradigm. • Students should be able to translate a real world problem to object oriented model. • Student can familiar with encapsulation, abstraction, inheritance, and polymorphism concepts.
Syllabus	Modular vs. Object-Oriented Paradigm, Abstraction, Encapsulation, Information hiding; Classes and Objects with C++: Data members, Member functions, Public/private access, Constructors, Destructors, Overloaded constructors, Constant member functions, Arrays of objects (both static and dynamic), Objects as arguments, Returning objects from functions, Copy constructor, Pointers as member variables, Shallow copy vs. Deep copy, Destructor, this pointer, Constant member variables, Constant objects, Static member variables, Static member functions; Operator Overloading: Simple binary operators, Overloading assignment operator for classes with dynamic memory allocation, Overloading logical and unary operators, Friend functions, Overloading operators as friend functions, Overloading stream insertion and extraction operators, Some other operators; I/O and File Processing: Text filing, Binary filing; Aggregation and Composition: Classes within classes, UML, Constructor, destructor calling sequence; Inheritance: Basics, Examples, UML, Public inheritance, Protected access specifier, Public inheritance vs. private and protected inheritance, Multiple inheritance, Diamond problem, Virtual inheritance, Writing copy constructors and overloading assignment operator for derived classes; Inheritance and Polymorphism: Virtual functions, Static vs. dynamic binding, Pure virtual functions, Abstract classes, Examples;

	Templates: Function templates, Class templates; Exceptions and Exception handling; Recursion: Basics, Examples
Contents	1. Introduction to Object Oriented Concepts 1.1. Real world examples Define the keyword ‘class’ 2.1. Access modifiers 2.2. Setter/Mutator and Getter/Accessor methods 2.3. Constructor & Destructor 3. Pointer/Reference to objects Preventing changes in data members from a method 4.1. Constant method 4.2. Constant data members Static functions Static data members constant and static objects 8. Calling sequence of Constructor & Destructor for constant and static objects Composition/Aggregation Cont... Nameless objects 11. Array of objects; 12. Operator Overloading 13. Friend functions Inheritance 14.1. Multilevel Inheritance 14.2. Private Inheritance 15. Polymorphism 15.1. Pure virtual functions and abstract class Diamond inheritance 17. Virtual inheritance

<table>
<tr><td></td><td>18. Template
18.1. Templated Function
18.2. Templated Class
19. Exception handling</td></tr>
<tr><td>Teaching-learning Strategies</td><td>• Interactive class session
• Hands on practices in class
• Brainstorming and Group discussion sessions
• Coding in LABS</td></tr>
<tr><td>Assignments</td><td>Coding Assignments 5</td></tr>
<tr><td>Assessment and Examinations</td><td>
<table>
<tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.</td></tr>
</table>
</td></tr>
</table>

Textbooks	• **A.** Deitel, H. M., & Deitel, P. J. (2010). C++ How to Program 6 th Edition. Prentice Hall. • **B.** Gaddis, T., & Sengupta, P. (2012). Starting Out with C++: From Control Structures Through Objects. Pearson.
Reference Material/Suggested Readings	• **R1.** Handouts. • **R2.** Shtern, V. (2000). Core C++: A software engineering Approach. Prentice Hall., • **R3.** Prata, S. (2002). C++ primer plus. Sams Publishing**.** • **R4.** Stroustrup, B. (2013). The C++ Programming Language

Detailed Lecture wise plan

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
1	1	Course Introduction Overview/Extension of/to Programming Fundamentals • Function Atomicity (Cohesion/Coupling) Pointer/Alias	B (9,10)	
	2	Overview/Extension of/to PF Cont... Pointer/Alias, Arrays, Dynamic Memory Allocation	B (9,10)	
2	3	Overview/Extension of/to PF Cont... • C structs: Data Driven Programming ○ Use/Benefits of Data Driven Programming struct keyword	B (11)	**Quiz#1**
	4	Overview/Extension of/to PF Cont... • C structs: Data Driven Programming ○ Struct as other struct members Array of structs	B (11)	
3	5	Overview/Extension of/to PF Cont... • C structs: Data Driven Programming struct objects on heap	B (11)	**Quiz#2** **Assign-1**
	6	Overview/Extension of/to PF Cont... • PF constructs: enum, union Introduction to Object Oriented Concepts and Terminology: Real world examples	R2-(Ch-1 – Remedy-3) R2-(Ch-8)	
4	7	Define the keyword ‘class’ Access modifiers: private and public only; First Member function; Data Abstraction through Encapsulation; Setter/Mutator and Getter/Accessor methods;	A-(Ch-3 (3.1~3.6))	

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
	8	Two Special Method: Constructor & Destructor; Constructor with No argument (Default Constructor); Constructor with Default arguments Constructor Delegation	A-(Ch-3 (3.7~3.11)) A-(Ch-9 (9.1 ~ 9.6))	**Quiz#3**
5	9	Pointer as data member Pointer/Reference to objects, Passing objects to functions by reference Pointer this Importance of destructor Calling sequence of Constructor & Destructor for multiple objects	A-(Ch-9 (9.7 ~ 9.12)) A-(Chapter-10.5, 10.6)	
	10	Passing objects to functions by value, Default Member-wise copy (Assignment & Initialization), Problem of Member-wise copy & its solution: Copy Constructor, Calling sequence of Constructor & Destructor: when objects are passed by value/reference	R1	**Quiz#4**
6	11	Preventing changes in data members from a method: const method; Constant data members Static functions Static data members constant and static objects Calling sequence of Constructor & Destructor for constant and static objects	A-(Chapter 10.2, 10.7)	
	12	Object as data member (Composition), Aggregation; Cascading calls with and without this pointer.	A-(Chapter 10.3) R2-(Chapter 12)	**Quiz#5**

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
7	13	Composition/Aggregation Cont... Nameless objects Array of objects;	A-(Chapter 10.3) R2-(Chapter 12)	
	14	Operator Overloading; Binary Operator receiving Instance of class as 1st operand	A-(Chapter 11.7)	
8	15	Overloading Unary Operator: as member, as non-member Unusual Operators: ++, --,	A-(Chapter 11.11)	**Quiz#6**
	16	[ ] (set & get, both versions), type-cast, Parentheses	A-(Chapter 11.8, 11.9)	**Assign-2**
9	17	Friend functions (efficient but shake the concept of encapsulation); Declaring a global function as friend of a class; Declaring member function of a class as friend of another class Friend class (an easy but more un-secure way);	A-(Chapter 10.4)	
	18	Operator NOT receiving Instance of class as 1st operand Overloading Binary Operators (Stream Insertion/Extraction) for I/O stream	A-(Chapter 11.5)	**Quiz#7**
10	19	In theory what is inheritance is-A Relationship: Public Inheritance Protected data member protected, private inheritance	A-(Chapter 12.1~12.4) R2-(Chapter 14)	
	20	Multilevel Inheritance: Direct and Indirect Base Class Calling of Constructor and Destructor for Derived Class Objects; Explicit call to the constructor of Base class from Derived class;	A-(Chapter 12.5, 12.6) Case Study-A	**Assign-3**

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
11	21	Review of Simple Inheritance, Multi Level Inheritance, Multiple Inheritance A review of OO relationships terminologies: Aggregation, composition, generalization (is-A), knows-a.	A-(Chapter 12) Case Study-B	**Quiz#8**
	22	Polymorphism: Overriding base-class members in derived class; Virtual functions and Dynamic binding; Concept of v-Pointer and v-Table	A-(Chapter 13) R1-Reading Material Case Study-B	
12	23	Pure virtual functions and abstract class; Defining a pure virtual destructor	R1-Reading Material	**Assign-4**
	24	Detail discussion on v-table structure, virtual constructor, object cloning	R1-Reading Material	
13	25	Some tricky things in pointer type casting and applying them on different topics studied so far.	R1-Reading Material	**Quiz#9**
	26	Diamond inheritance Virtual inheritance	R1-Reading Material	
14	27	C++ Streams, Members and Manipulators of Streams; File Handling using Streams	R1-Reading Material A-(Chapter Chapter-15)	**Quiz#10**
	28	Access Techniques: Sequential, Direct, and Random Access Files; Input/Output of Object from/to File (binary/ text mode);	R1-Reading Material A-(Chapter Chapter-17)	
15	29	Function Template; Overloading of Function Template; Specialized of Template Function	A-(Chapter 14.2,14.3)	**Assign-5**
	30	Class Template; Specialized method of Template Class; Complete Specialized Template Class	A-(Chapter 14.4, 14.5)	**Quiz # 11**

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
16	31	Friendship and Inheritance with Templates	A-(Chapter 14.6, 14.7, 14.8)	
	32	What is Exception? Error vs. Exception; Evolution of Exception Handling: exit, abort, assert, new-keywords; try, catch, throw Unhandled Exception; Propagation of Exception and its advantage	A-(Chapter 16)	

<table>
<tr><td>Program</td><td colspan="2">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="2">CC-215</td></tr>
<tr><td>Course Title</td><td colspan="2">Database Systems</td></tr>
<tr><td rowspan="2">Credit Hours</td><td>Theory</td><td>Lab</td></tr>
<tr><td>3</td><td>1</td></tr>
<tr><td>Lecture Duration</td><td colspan="2">90 minutes (1.5 Hours), 2 lectures per week, 3 hours lab session per week</td></tr>
<tr><td>Semester</td><td colspan="2">3</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td>Courses</td><td>Knowledge</td></tr>
<tr><td>Nil</td><td>Nil</td></tr>
<tr><td>Follow Up Courses</td><td colspan="2">Advanced Database Management System</td></tr>
<tr><td>Aims and Objectives</td><td colspan="2">Upon completion of this course, the student will be able to:
1. Grasp the theory of database design
2. Should be able to design a database
3. Should be able to write queries</td></tr>
<tr><td colspan="3">Course Learning Outcomes (CLOs)</td></tr>
<tr><td>CLO No</td><td>Course Learning Outcome</td><td>Bloom Taxonomy</td></tr>
<tr><td>CLO-1</td><td>Explain fundamental database concepts.</td><td>C2 (Explain)</td></tr>
<tr><td>CLO-2</td><td>Design conceptual, logical and physical database schemas using different data models.</td><td>C5 (Design)</td></tr>
<tr><td>CLO-3</td><td>Identify functional dependencies and resolve database anomalies by normalizing database tables.</td><td>C2 (Identify)</td></tr>
<tr><td>CLO-4</td><td>Use Structured Query Language (SQL) for database definition and manipulation in any DBMS</td><td>C4 (Use)</td></tr>
<tr><td>Learning Outcomes</td><td colspan="2">At the end of the course, you should be able to:
• Explain fundamental database concepts.
• Design conceptual, logical and physical database schemas using different data models.</td></tr>
</table>

	• Identify functional dependencies and resolve database anomalies by normalizing database tables. • Use Structured Query Language (SQL) for database definition and manipulation in any DBMS
Contents	1. Introduction 1.1. Basic database concepts 1.2. Database approach 1.3. File based system 2. Database architecture 2.1. Three level schema architecture 2.2. Data independence 3. Relational data model 3.1. Attributes 3.2. Schemas 3.3. Tuples 3.4. Domains 3.5. Relation instances 4. Details of relational data model 4.1. Keys of relations 4.2. Integrity constraints 5. Relational algebra 5.1. Selection 5.2. Projection 5.3. Cartesian product 5.4. Types of joins 6. Normalization 6.1. Functional dependencies 6.2. Normal forms 7. Entity relationship model

<table>
<tr><td></td><td>7.1. Entity sets
7.2. Attributes
7.3. Relationship
8. Structured Query Language (SQL)
8.1. Joins
8.2. Sub-queries in SQL
8.3. Grouping SQL
8.4. Aggregation in SQL
9. Concurrency control
9.1. Database backup
9.2. Recovery
9.3. Indexes
9.4. NoSQL systems.</td></tr>
<tr><td>Teaching-learning Strategies</td><td>• Interactive class session
• Hands on practices in class
• Brainstorming and Group discussion sessions</td></tr>
<tr><td>Assignments</td><td>• Paper based written assignments 3
• Project 2
• Quiz 4</td></tr>
<tr><td>Assessment and Examinations</td><td><table>
<tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
</table></td></tr>
</table>

	2	Midterm Assessment	35%	It takes place at the mid-point of the semester.
	3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper.
Textbooks & Reference material	• Database Systems: A Practical Approach to Design, Implementation, and Management, 6th Edition by Thomas Connolly and Carolyn Begg • Database Systems: The Complete Book, 2nd Edition by Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer Widom • Database System Concepts, 6th Edition by Avi Silberschatz, Henry F. Korth and S. Sudarshan. • Database Management Systems, 3rd Edition by Raghu Ramakrishnan, Johannes Gehrke			
Notes	• The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester.			

Program	BS Data Science		
Course Code	CC-213		
Course Title	Data Structures		
Credit Hours	**Theory**	**Lab**	
	3	1	
Lecture Duration	90 minutes (1.5 Hours), 2 lectures per week, 3 hours lab session per week		
Semester	3		
Pre-requisites	**Courses**	**Knowledge**	
	Programming Fundamentals	Nil	
Follow Up Courses	Operating Systems, Analysis of Algorithms		
Course Learning Outcomes (CLOs):			
CLO No	**Course Learning Outcome**	**Bloom Taxonomy**	
CLO-1	Implement various data structures and their algorithms and apply them in implementing simple applications	C3 (Apply)	
CLO-2	Analyze simple algorithms and determine their complexities.	C5 (Analyze)	
CLO-3	Apply the knowledge of data structure to other application domains.	C3 (Apply)	
CLO-4	Design new data structures and algorithms to solve problems.	C6 (Design)	
Aims and Objectives	1. To introduce data structures as basic building blocks of large programs. 2. To learn the commonly used data structures. 3. To introduce the notion of time and space complexity. 4. To develop the skills to analyze time and space requirements for a data structure and associated algorithms.		

	5. To prepare the students to pick the right data structure for a given problem.
Learning Outcomes	• Implement various data structures and their algorithms and apply them in implementing simple applications • Analyze simple algorithms and determine their complexities. • Apply the knowledge of data structure to other application domains. • Design new data structures and algorithms to solve problems.
Syllabus	Abstract data types, complexity analysis, Big Oh notation, Stacks (linked lists and array implementations), Recursion and analyzing recursive algorithms, divide and conquer algorithms, Sorting algorithms (selection, insertion, merge, quick, bubble, heap, shell, radix, bucket), queue, dequeuer, priority queues (linked and array implementations of queues), linked list & its various types, sorted linked list, searching an unsorted array, binary search for sorted arrays, hashing and indexing, open addressing and chaining, trees and tree traversals, binary search trees, heaps, M-way tress, balanced trees, graphs, breadth-first and depth-first traversal, topological order, shortest path, adjacency matrix and adjacency list implementations, memory management and garbage collection.
Contents	1. Collections, Abstract data types, Complexity analysis, Big Oh notation 2. Recursion and analyzing recursive algorithms, divide and conquer algorithms 3. Sorting algorithms (selection, insertion, merge, quick, bubble, heap, shell, radix, bucket) 4. Array vs. Linked representation of Collections 5. Stacks (linked lists and array implementations)

<table>
<tr><td></td><td>
6. Queue (linked lists and array implementations), Introduction to priority queues
7. Lists (linked and array implementations), Various types of linked lists, sorted linked list
8. Trees and tree traversals, binary search trees, heaps, M-way tress, balanced trees
9. Heaps and priority queues
10. Graphs, breadth-first and depth-first traversal, topological order, shortest path, adjacency matrix and adjacency list implementations
11. Searching an unsorted array, binary search for sorted arrays, hashing and indexing, open addressing and chaining
• Memory management and garbage collection.
</td></tr>
<tr><td>Teaching-learning Strategies</td><td>
• Interactive class session
• Hands on practices in class
• Brainstorming and Group discussion sessions
• Coding in LABS
</td></tr>
<tr><td>Assignments</td><td>
• Paper based written assignments 4
• Coding assignments 6
</td></tr>
<tr><td>Assessment and Examinations</td><td>
<table>
<tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
</table>
</td></tr>
</table>

	3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.
Textbooks	• Data Structures and Algorithm Analysis in C++ by Mark Allen Weiss			
Reference Material/Suggested Readings	• Data Structures and Algorithms in C++ by Adam Drozdek • Data Structures and Algorithm Analysis in Java by Mark A. Weiss • Data Structures and Abstractions with Java by Frank M. Carrano & Timothy M. Henry • Java Software Structures: Designing and Using Data Structures by John Lewis and Joseph Chase			
Notes	• Academic integrity is expected of all students. Plagiarism or cheating in any assessment will result in at least an F grade in the course, and possibly more severe penalties. • You bear all the responsibility for protecting your assignments from plagiarism. If anyone else submits your assignment or uses your code in his/her assignment, you will be considered equally responsible. • The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester. • Introductory knowledge of using the computers is assumed for this course. All code written in quizzes, assignments, homework's, and exams must be in JavaScript. Code must be intelligently			

	documented (commented). Undocumented code may not be given any credit. • The IDE use is not allowed, Notepad++ has to be used for coding. • There is no makeup for a missed sessional grading instruments like quizzes, assignments, and homework's.

Detailed Lecture wise plan

Week	Lecture	Topic	SourceBook	**Recommendation for Learning Activities**
1	1	Introduction to Data Structures; Role of Data Structures in Computer Science Defining Algorithm: Properties of Algorithm		
	2	Introduction to Algorithm's Performance Analysis and Measurement Learning to Calculate Running Time of Different Code Snippets, Examples i.e. Binary search, Selection sort etc;		
2	3	More on Step Counting (Big Oh Notation)		
	4	Case Study: Polynomial as ADT: Take it as sample application to decide its structure and operations and also calculating the step counting of its operations.		
3	5	(Arrays)Matrix, Row major and column major Representation of N-Dimensional Arrays in different Languages.		
	6	Sparse Matrices		
4	7	The Stack ADT, Applications of Stack: Function Call Stack, Usage of Stack in different CS Applications.		
	8	Application of Stack: Expressions Evaluation		
5	9	Queues: Linear/Circular, Applications of Queue.		
	10	Recursive Definition and Processes, Direct Recursion, Learning the Recursive Trace		
6	11	Recursion Continued: Binary Search, Exiting from Maze, Towers of Hanoi and Islamic Fractals as an example		
	12	Recursion Continued:		
7	13	Review of Dynamic Memory Allocation; Object Manipulation of Self Referential objects		
	14	Linear Single Link List Linked Stacks/Queues Linear Double Link List		

Week	Lecture	Topic	SourceBook	**Recommendation for Learning Activities**
8	15	Circular Single Link List, Circular Double Link List Container vs Iterator: Defining Iterator for Link List		
	16	Array-based implementation of Link-based Structures, Generalized Lists		
Midterm Exam				
9	17	Introduction to Trees, Tree Terminology, Logical construction and Representation of Trees, Introduction to Binary Tree ADT, Mathematical properties Tree Traversals Array-Based Implementation of Binary Trees (Insertion and Traversing)		
	18	Linked Implementation of Binary Trees (Insertion, Traversing, Searching and deletion in Binary Trees)		
10	19	Linked Implementation of Binary Trees Continued:		
	20	Binary Search Tree: Mathematical Properties and its implementation		
11	21	Height Balance Trees: AVL Tree: Insertion in AVL		
	22	Deletion Operation in AVL		
12	23	Heaps (MinHeap and MaxHeap) Heaps as Priority Queues		
	24	Heap continued: (Min-Max Heap, Deaps)		
13	25	Introduction to graph and related terminology Representation of Graphs Elementary Graph Operations, DFS, BFS		
	26	Spanning Trees Connectivity in Graphs		
14	27	Hashing and Overflow Handling		
	28	Hashing continued…		
15	29	Introduction to Sorting types and Techniques, Logical and Algorithmic Implementation of Bubble, Insertion, Selection, Merge, and Quick Sort		

Week	Lecture	Topic	SourceBook	**Recommendation for Learning Activities**
	30	Sorting Continued…		
16	31	Balanced Search Trees: Theoretical Comprehension of Insertion/Deletion Operations in Balanced-Search Trees; 2-3:Tree insertion		
	32	Balanced Search Trees cont…: 2-3 Tree Deletion		
Final Exam				

<table>
<tr><td>Program</td><td colspan="3">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="3">CC-311</td></tr>
<tr><td>Course Title</td><td colspan="3">Operating Systems</td></tr>
<tr><td rowspan="2">Credit Hours</td><td colspan="2">Theory</td><td>Lab</td></tr>
<tr><td colspan="2">3</td><td>1</td></tr>
<tr><td>Lecture Duration</td><td colspan="3">90 minutes (1.5 Hours), 2 lectures per week, 3 hours lab session per week</td></tr>
<tr><td>Semester</td><td colspan="3">5</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td colspan="2">Courses</td><td>Knowledge</td></tr>
<tr><td colspan="2">Data Structures and Algorithms</td><td></td></tr>
<tr><td>Follow Up Courses</td><td colspan="3">System Programming</td></tr>
<tr><td colspan="4">Course Learning Outcomes (CLOs)</td></tr>
<tr><td>CLO No</td><td colspan="2">Course Learning Outcome</td><td>Bloom Taxonomy</td></tr>
<tr><td>CLO-1</td><td colspan="2">Acquire the basic knowledge of computer organization computer architecture and assembly language.</td><td>C2 (Understand)</td></tr>
<tr><td>CLO-2</td><td colspan="2">Understand the concepts of basic computer organization, architecture, and assembly language techniques</td><td>C2 (Understand)</td></tr>
<tr><td>CLO-3</td><td colspan="2">Solve the problems related to computer organization and assembly language</td><td>C3 (Apply)</td></tr>
<tr><td>Aims and Objectives</td><td colspan="3">1. To understand the internals of operating system and practically access its services to have a clear understanding of the working of OS Kernel</td></tr>
</table>

Learning Outcomes	• Understand the characteristics of different structures of the Operating Systems and identify the core functions of the Operating Systems (Understand) • Analyze and evaluate the algorithms of the core functions of the Operating Systems and explain the major performance issues with regard to the core functions (Evaluate) • Demonstrate the knowledge in applying system software and tools available in modern operating systems (Demonstrate)
Syllabus	Operating systems basics, system calls, process concept and scheduling, inter-process communication, multithreaded programming, multithreading models, threading issues, process scheduling algorithms, thread scheduling, multiple-processor scheduling, synchronization, critical section, synchronization hardware, synchronization problems, deadlocks, detecting and recovering from deadlocks, memory management, swapping, contiguous memory allocation, segmentation & paging, virtual memory management, demand paging, thrashing, memory-mapped files, file systems, file concept, directory and disk structure, directory implementation, free space management, disk structure and scheduling, swap space management, system protection, virtual machines, operating system security
Contents	**Section 1:** - Introduction - Intro to Linux Environment - Program v/s Process - Process Management **Section 2:** - I/O Redirection and IPC - Thread Management **Section 3:** - Introduction to Synchronization - S/W-based and H/W-based CSP Solutions - Synchronization using Semaphore - Synchronization using Monitor - Deadlocks **Section 4:** - Memory - Paging

<table>
<tr><td></td><td>- Virtual Memory
<u>Section 5:</u>
- Disk Geometry and Partitioning
- Disk Formatting and File System Monitoring
- File-System Architecture
- File Permissions</td></tr>
<tr><td>Teaching-learning Strategies</td><td>• Lectures
• Case Studies
• Project
• Assignments</td></tr>
<tr><td>Assignments</td><td>Types and Number with calendar</td></tr>
<tr><td>Assessment and Examinations</td><td>
<table>
<tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.</td></tr>
</table>
</td></tr>
<tr><td>Textbooks</td><td>A. Operating System Concepts, by Galvin, Gagne, 10th Edition Silberschatz, Published in 2019, ISBN- 978-1-118-06333-0</td></tr>
</table>

Reference Material/Suggested Readings	B. Modern Operating Systems, by Andrew S. Tanenbaum, 4th edition, Published in 2016, ISBN- 9789332575776 C. Operating Systems, Internals and Design Principles, by William Stallings, 9th edition,Published in 2017, ISBN-13: 978-0134670959 D. Dr. Muhammad Arif Butt, OS -Video Lectures: https://www.youtube.com/c/LearnWithArif/playlists
Notes	

Detailed Lecture wise plan

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
1	1	**Introduction to course**, pre-requisite, policies, tools, and grading system. What is an Operating System (OS) and why it is needed to manage h/w? Operating System services, interrupts, traps and signals. Dual mode operations and protection mechanism. Types of operating systems and computing environments	Text A-Ch1	
	2	**Introduction to virtualization and hypervisors**. Installing Linux (Ubuntu, Kali, CentOS) on Virtualbox. Introduction to Linux command line interface, Linux File Hierarchy Standard and basic shell commands. Linux system call interface. Compiling a C program on Linux command line interface	Text A-Ch1	
2	3	**Editors used in Linux** (vim, peco, nano). Shell commands (wc, sort, uniq, grep, cut, paste, comm, comp, diff, whereis, which, locate, find, tar, gzip, gunzip). Program on disk and its components. Viewing contents of a program file. Process in memory and its components (stack, heap and PCB). Command line arguments and environment variables. Viewing contents of a running program using readelf and objdump	Handouts	
	4	**CPU and I/O bound processes**. Process state models. Five, six and seven state process models. Process scheduling queues. Long term, medium term and short term schedulers. Concept of process/context switch. Process creation and termination. Shell commands related to process management. Fork, wait and exit system calls. Interrupt, trap and system calls. Process resource limits.	Text A-Ch2	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
3	5	**Discussion on working of a Linux shell** and the concept of how an internal and external command executes. Running programs in the background and foreground. Switching programs to different states. Basic commands related to process states like ps, fg, bg and top.	Text A-Ch3	
	6	**The open, read, write and close** paradigm in Linux. The concept of PPFDT. The connection of an opened file from process PPFDT to System Wide File Table, to I-node table and finally to disk blocks. Cooperating Processes. Taxonomy of Inter-process Communication	Handouts	**Lab:**
4	7	**I/O Redirection, UNIX IPC tools.** Using pipes, FIFOs and signals in Linux	Text A-Ch2	
	8	**Concurrent and parallel programming**. Introduction to threads. Multi-threading. Merits and demerits of threads. User level vs Kernel level threads. Threading models. Programming using Pthread library.	Text A-Ch4	**Lab:**
5	9	**Process scheduler and Dispatcher.** Preemptive vs non-preemptive scheduling. CPU and IO bursts. CPU scheduling and scheduling criteria, FCFS, SJF, SRTF, and Priority scheduling.	Text A Ch6	
	10	**Round Robin,** Virtual Round Robin, Multi level Queue Scheduling and Multi level Feed- back queue scheduling, Rotating Stair-case Dead line scheduler,	Text A-Ch6	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
6	11	**Rotating Stair-case dead line scheduler,** UNIX SVR3 scheduling algorithm. Changing process priorities using nice, renice commands. Changing nice value of running processes and executing a program with a nice value other than the default. The concept of hard and soft CPU affinity in Linux. Displaying and changing the scheduling parameters of Linux processes using schedtool	Handouts	
	12	**Introduction to synchronization,** Concurrency Control, Race Condition, Critical Section Problem. Concept of atomic operation. General format of a CS problem solution. Characteristics of a good CSP solution.	Text A-Ch5	
7	13	**Software Based Solutions to CSP:** Dekker solution, Peterson solution, and Leslie Lamport's Bakery algorithm. Concept of busy waiting.	Text A-Ch5	
	14	**H/W based solutions to CSP:** Disabling of interrupts, TSL and swap instructions. Thread synchronization using pthread_mutex_t variable and pthread_mutex_lock() and pthread_mutex_unlock() library calls	Text A-Ch5	
8	15	**Introduction to semaphores.** Binary and counting semaphores. Achieving mutual exclusion using semaphores. Achieving serialization using semaphores. Solution to Standard Synchronization problems using semaphores, Producer Consumer, Dining Philosopher, Reader writer, Sleeping Barber, Smokers problem	Text A-Ch5	
	16	**Limitations of semaphores,** Introduction to Monitors, Condition variables, Hoare and Mesa monitors. Solution to standard synchronization problems using monitors.	Text A-Ch5	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
9	17	**Introduction to Dead locks,** Four necessary and sufficient conditions for Dead Locks, Resource allocation graph, Dead lock handling methods, Dead lock prevention	Text A-Ch7	
	18	**Dead Lock Avoidance.** Bankers and Safety Algorithm. Dead Lock Detection and Recovery Algorithms	Text A-Ch7	
10	19	**Memory management,** address binding and linking, Logical vs Physical addresses, Dynamic loading, Dynamic linking and shared libraries. Overlays, swapping. Introduction to contagious memory allocation	Text A-Ch8	
	20	**MFT and MVT,** Placement algorithms, Internal and External fragmentation, Buddy partitioning scheme.	Text A-Ch8	
11	21	**Introduction to paging,** Page Table, Address translation in paging, Paging parameters for Intel and PDP11	Text A-Ch8	
	22	**Implementing page table in cache,** memory and CPU registers, Structure of Page Tables (Hierarchical, Inverted and Hashed Page tables). Introduction to Segmentation, address translation in segmentation.	Text A-Ch9	
12	23	**Introduction to paged segmentation,** address translation in a paged segmentation. Address translation in Intel 80386 (Real and protected mode)	Text A-Ch9	
	24	**Virtual Memory,** Background, Demand Paging, Performance of Demand Paging, Page Replacement algorithms (FIFO, Optimal, LRU, LFU, MFU, Buffering)	Text A-Ch9	
13	25	**Copy on Write protocol and vfork() system call,** Allocation Of Frames, Thrashing, Resident Set Management, Working Set Model, Page fault frequency, memory mapped files.	Text A-Ch9	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
	26	**Hard Disk Geometry:** Spinning and Solid state disk. Working of spinning disk and its interfaces (IDE, ATA, SATA, SCSI, SAS). The concept of Logical Block Addressing and its mapping on CHS address. **Hard Disk Partitions:** Partitioning a hard disk. Different types of partition tables, MBR and GPT. Linux tools used for partitioning a hard disk like fdisk, gdisk, parted, gparted, cfdisk, sfdisk	Text A-Ch10	
14	27	**Disk Formatting:** Concept of a file system and the basic functionalities that every file system should offer. Comparison of different file systems like ext2/3/4, reiserfs, hpfs, minix, ntfs, vfat, xfs and zfs. Use of Linux tools like mkfs, mke2fs, mkntfs, mkfs.fat, mkfs.minix to put a file system on a partition **File System Mounting:** Introduction to the concept of file system mounting. Linux configuration files related to file system mounting. Linux commands like mount, umount, lsblk, blkid. Maintaining integrity of file system using Linux commands like fsck, e2fsck, fsck.fat, fsck.nfs	Text A-Ch11	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
	28	**File System Architecture:** Schematic view of a standard UNIX file system. Describe the contents of boot block, super block, inode block, and data blocks. Discuss In-memory and on-disk structures used by a file system. Describes what actually happens behind the curtain when a user creates, accesses and deletes a file and how Linux keep track of opened files by a process. Use of Linux commands like df, du, lsof, fuser, and tune2fs to perform these tasks **Hard and Soft Links:** Discuss the use of hard and soft links on all UNIX based systems. Differences between hard and soft links. Use of Linux command ln to create hard and soft links	Text A-Ch12	
15	29	**File Permissions:** Discuss the use of standard file permissions. How to change the existing file permissions on a file using symbolic and octal way. Use of chmod and chown commands. Setting the default file permissions on a newly created file using the umask command **Special File Permissions:** Concept and use of Saved SUID bit on files. Concept and use of Saved SGID bit on files and directories. Concept and use of Sticky bit on files and directories.	Text A-C14	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
	30	**Access Control Lists:** Discuss the security on files using Access Control List. Concept of Discretionary Access control and Mendatory Access control. How to set ACLs on files. A discussion on default ACLs or ACLs on directories **Device Files:** Seven File Types in Linux and the concept of device files. Describes the contents of /dev/ directory. Describes Major and minor numbers and shows how you can create your own device files. Important /dev/ directory files like zero, null, full, random, urandom and ttys	Handouts	
16	31	**Terminal Attributes:** Overview of Terminal Devices and a comparison between disk and terminal files. Examine current attributes of terminal driver on a Linux machine and changing them using stty command. Overview of Canonical and Non-canonical mode of terminal drivers. **Time Management in Linux operating system** **Managing services using systemd:** Introduction to Linux system daemon. Overview of to systemd unit files, specially Target Unit Files and Service Unit Files. Shell commands to manage services using systemctl. Writing/running a basic service of your own	Handouts	
	32	**Log Files:** Logging mechanism in Linux Booting process of a Linux system: A discussion on five phases of Linux Operating system: BIOS / UEFI Initialization, Master Boot Record, Boot Loader, Kernel Initialization, init or systemd Process	Text A-Ch18	

<table>
<tr><td>Program</td><td colspan="2">BS data science</td></tr>
<tr><td>Course Code</td><td colspan="2">CC-303</td></tr>
<tr><td>Course Title</td><td colspan="2">Software Engineering</td></tr>
<tr><td rowspan="2">Credit Hours</td><td>Theory</td><td>Lab</td></tr>
<tr><td>3</td><td>0</td></tr>
<tr><td>Lecture Duration</td><td colspan="2">90 minutes (1.5 Hours), 2 lectures per week</td></tr>
<tr><td>Semester</td><td colspan="2">4</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td>Courses</td><td>Knowledge</td></tr>
<tr><td>Nil</td><td>Nil</td></tr>
<tr><td>Follow Up Courses</td><td colspan="2"></td></tr>
<tr><td>Aims and Objectives</td><td colspan="2">Upon completion of this course, the student will be able to:
1. Understanding Grasp modeling concepts with emphasis on performance analysis.
2. Planning of software
3. Designing Software</td></tr>
</table>

Course Learning Outcomes (CLOs):

CLO No	Course Learning Outcome	Bloom Taxonomy
CLO-1	Describe various software engineering processes and activates techniques for the design of digital electronic circuits	C1 (Describe)
CLO-2	Apply the system modeling techniques to model a medium size software systems	C3 (Apply)
CLO-3	Apply software quality assurance and testing principles to medium size software systems	C4 (Apply)
CLO-4	Discuss key principles and common methods for software project management such as scheduling, size estimation, cost estimation and risk analysis	C2 (Discuss)

Learning Outcomes	At the end of the course, you should be able to: • Describe various software engineering processes and activates

	• Apply the system modeling techniques to model a medium size software systems • Apply software quality assurance and testing principles to medium size software systems • Discuss key principles and common methods for software project management such as scheduling, size estimation, cost estimation and risk analysis
Contents	1. Describe various software engineering processes and activates 1.1. Nature of Software 1.2. Overview of Software Engineering 1.3. Professional software development 2. Software engineering practice 2.1. Software process structure 2.2. Software process models 2.3. Agile software Development 2.4. Agile process models 2.5. Agile development techniques 3. Requirements engineering process 3.1. Functional requirements 3.2. Non-functional requirements 4. Model driven engineering 4.1. Context models 4.2. Interaction models 4.3. Structural models 4.4. Behavioral models 5. Architectural design 5.1. Design and implementation 6. UML diagrams 6.1. Design patterns 6.2. Software testing and quality assurance

	6.3. Software evolution 6.4. Project management 6.5. Project planning 7. Configuration management 7.1. Software Process improvement
Teaching-learning Strategies	• Interactive class session • Hands on practices in class • Brainstorming and Group discussion sessions
Assignments	• Paper based written assignments 3 • Project 2 • Quiz 4

Assessment and Examinations

Sr. #	Elements	Weightage	Details
1	Formative Assessment	25%	It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.
2	Midterm Assessment	35%	It takes place at the mid-point of the semester.
3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper.

Textbooks &	• Software Engineering, Sommerville I., 10th Edition, Pearson Inc., 2014

Reference material	• Software Engineering, A Practitioner's Approach, Pressman R. S.& Maxim B. R., 8th Edition, McGraw-Hill, 2015.
Notes	• The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester.

Program	BS Data Science	
Course Code	CC-214	
Course Title	Computer Networks	
Credit Hours	**Theory**	**Lab**
	3	1
Lecture Duration	90 minutes (1.5 Hours), 2 lectures per week, 3 hours lab session per week	
Semester	6	
Pre-requisites	**Courses**	**Knowledge**
	None	
Follow Up Courses		
Course Learning Outcomes (CLOs)		

CLO No	**Course Learning Outcome**	**Bloom Taxonomy**
CLO-1	Describe the key terminologies and technologies of computer networks	C2 (Describe)
CLO-2	Explain the services and functions provided by each layer in the Internet protocol stack.	C2 (Explain)
CLO-3	Identify various internetworking devices and protocols and their functions in a networking	C4 (Identify)
CLO-4	Analyze working and performance of key technologies, algorithms and protocols	C4 (Analyze)
CLO-5	Build Computer Network on various Topologies	C3 (Build)

Aims and Objectives	The course will cover computer networks in a top down manner starting from the application layer to data link layer. The course will be

	taught in the Internet perspective and will therefore cover the layers of the TCP/IP suite. After the completion of this course, students will be able to: 1. Understand the fundamental concepts of networking. 2. Know the working of each layer in TCP/IP suite. 3. Identify the challenges involved in data flow and error control. 4. understand the working of internet.
Learning Outcomes	5. CLO-1: Describe the key terminologies and technologies of computer networks 6. CLO-2: Explain the services and functions provided by each layer in the Internet protocol stack. 7. CLO-3: Identify various internetworking devices and protocols and their functions in a networking 8. CLO-4: Analyze working and performance of key technologies, algorithms and protocols 9. CLO-5: Build Computer Network on various Topologies
Syllabus	Introduction and protocols architecture, basic concepts of networking, network topologies, layered architecture, physical layer functionality, data link layer functionality, multiple access techniques, circuit switching and packet switching, LAN technologies, wireless networks, MAC addressing, networking devices, network layer protocols, IPv4 and IPv6, IP addressing, sub netting, CIDR, routing protocols, transport layer protocols, ports and sockets, connection establishment, flow and congestion control, application layer protocols, latest trends in computer networks.
Contents	I. **Introduction** i. Overview of the Internet ii. Overview of Networking Protocols iii. Network Edge iv. Network Core v. Protocol Layers / Service Model vi. Physical Media vii. Some History of the Internet

	viii. General Networking Example II. **Application Layer** i. Principles of Networking Applications ii. Web and HTTP iii. FTP iv. Email v. DNS vi. Peer-to-Peer (P2P) III. **Transport Layer** i. Transport Layer Services ii. Multiplexing and Demultiplexing iii. Connectionless Transport: UDP iv. Principles of Reliable Data Transport v. Connection-Oriented Transport: TCP vi. Principles of Congestion Control vii. TCP Congestion Control IV. **Network Layer** i. Virtual Circuits and Datagram Networks ii. Inside a Router iii. Details of the Internet Protocol (IP) iv. IP Subnetting v. Routing Algorithms vi. Link State vii. Distance Vector viii. Hierarchical Routing V. **Link Layer** i. Error Detection and Correction ii. Multiple Access Protocols iii. Local Area Networks iv. Multiprotocol Label Switching (MPLS)) v. Data Center Networking
Teaching-learning Strategies	The course will be based on the following teaching and learning activities: - Lectures covering the theoretical part using PowerPoint presentations - Case studies - Review questions
Assignments	Total 4 Assignment

	Sr. #	Elements	Weightage	Details
Assessment and Examinations	1	Formative Assessment	25%	Assignments, Presentations, Quizzes.
	2	Midterm Assessment	35%	Mid Term exam of 90 Minutes in 9th week of the semester.
	3	Final Assessment	40%	End Term exam of 120-180 minutes at the end of semester.
Textbooks	• Kurose, J. F., & Ross, K. W. (2021). Computer networking: A top-down approach featuring the Internet. Boston: Addison-Wesley.			
Reference Material/Suggested Readings	• Tanenbaum, A. S. (2013. Computer networks. Upper Saddle River, N.J: Prentice Hall PTR. • Stallings, W. (2014). Data and computer communications. Upper Saddle River, N.J: Prentice Hall. • Forouzan, B. A., Coombs, C. A., & Fegan, S. C. (2012). Data communications and networking. Boston: McGraw-Hill.			
Notes	• Power Point slides with reading material from book.			

Detailed Lecture wise plan

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
1	1	What Is the Internet? A Nuts-and-Bolts Description A Services Description What Is a Protocol?	Ch-01	
	2	The Network Edge Access Networks Physical Media The Network Core Packet Switching Circuit Switching A Network of Networks	Ch-01	

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
2	3	Delay, Loss, and Throughput in Packet-Switched Networks Overview of Delay in Packet-Switched Networks Queuing Delay and Packet Loss End-to-End Delay Throughput in Computer Networks Protocol Layers and Their Service Models Layered Architecture Encapsulation Networks Under Attack	Ch-01	
	4	History of Computer Networking and the Internet The Development of Packet Switching: 1961–1972 Proprietary Networks and Internetworking: 1972–1980 A Proliferation of Networks: 1980–1990 The Internet Explosion: The 1990s The New Millennium	Ch-01	**Assignment-1**
3	5	Principles of Network Applications Network Application Architectures Processes Communicating	Ch-02	**Quiz-1**
	6	Transport Services Available to Applications Transport Services Provided by the Internet Application-Layer Protocols Network Applications	Ch-02	
4	7	The Web and HTTP Overview of HTTP Non-Persistent and Persistent Connections HTTP Message Format User-Server Interaction: Cookies Web Caching The Conditional GET	Ch-02	

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
	8	File Transfer: FTP FTP Commands and Replies Electronic Mail in the Internet SMTP Comparison with HTTP Mail Message Format Mail Access Protocols	Ch-02	
5	9	DNS—The Internet's Directory Service Services Provided by DNS Overview of How DNS Works DNS Records and Messages	Ch-02	
	10	Peer-to-Peer Applications P2P File Distribution Distributed Hash Tables (DHTs) Socket Programming: Creating Network Applications Socket Programming with UDP Socket Programming with TCP	Ch-02	**Assignment-2**
6	11	Introduction and Transport-Layer Services Relationship Between Transport and Network Layers Overview of the Transport Layer in the Internet Multiplexing and Demultiplexing	Ch-03	**Quiz-2**
	12	Connectionless Transport: UDP UDP Segment Structure UDP Checksum	Ch-03	
7	13	Principles of Reliable Data Transfer Building a Reliable Data Transfer Protocol Pipelined Reliable Data Transfer Protocols Go-Back-N (GBN) Selective Repeat (SR)	Ch-03	

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
	14	Connection-Oriented Transport: TCP The TCP Connection TCP Segment Structure Round-Trip Time Estimation and Timeout Reliable Data Transfer Flow Control TCP Connection Management	Ch-03	
8	15	Principles of Congestion Control The Causes and the Costs of Congestion Approaches to Congestion Control Network-Assisted Congestion- Control	Ch-03	
	16	TCP Congestion Control Fairness	Ch-03	
9	17	The Network Layer Introduction Forwarding and Routing Network Service Models Virtual Circuit and Datagram Networks Virtual-Circuit Networks Datagram Networks	Ch-04	**Assignment-3**
	18	What's Inside a Router? Input Processing Switching Output Processing Where Does Queuing Occur? The Routing Control Plane	Ch-04	**Quiz-3**
10	19	The Internet Protocol (IP): Forwarding and Addressing in the Internet Datagram Format IPv4 Addressing, Subnetting, CIDR Internet Control Message Protocol (ICMP) IPv6	Ch-04	

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
	20	Routing Algorithms The Link-State (LS) Routing Algorithm The Distance-Vector (DV) Routing Algorithm Hierarchical Routing	Ch-04	
11	21	Routing in the Internet Intra-AS Routing in the Internet: RIP Intra-AS Routing in the Internet: OSPF Inter-AS Routing: BGP	Ch-04	
	22	Broadcast and Multicast Routing Broadcast Routing Algorithms Multicast	Ch-04	
12	23	Introduction to the Link Layer The Services Provided by the Link Layer Where Is the Link Layer Implemented?	Ch-05	**Assignment-4**
	24	Error-Detection and -Correction Techniques Parity Checks Check summing Methods Cyclic Redundancy Check (CRC)	Ch-05	**Quiz-4**
13	25	Multiple Access Links and Protocols Channel Partitioning Protocols	Ch-05	
	26	Random Access Protocols Taking-Turns Protocols DOCSIS: The Link-Layer Protocol for Cable Internet Access	Ch-05	
14	27	Link-Layer Addressing and ARP Ethernet Link-Layer Switches	Ch-05	
	28	Virtual Local Area Networks (VLANs) Link Virtualization: A Network as a Link Layer Multiprotocol Label Switching (MPLS)	Ch-05	
15	29	Data Center Networking	Ch-05	
	30	A Day in the Life of a Web Page Request DHCP, UDP, IP, and Ethernet DNS and ARP	Ch-05	

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
16	31	Intra-Domain Routing to the DNS Server Web Client-Server Interaction: TCP and HTTP	Ch-05	
	32	Final Q & Answer session		

Program	BS Data Science	
Course Code	CC-308	
Course Title	Information Security	
Credit Hours	**Theory**	**Lab**
	3	0
Lecture Duration	90 minutes (1.5 Hours), 2 lectures per week	
Semester	8	
Pre-requisites	**Courses**	**Knowledge**
	None	Nil
Follow Up Courses	Nil	

Course Learning Outcomes (CLOs)

CLO No	Course Learning Outcome	Bloom Taxonomy
CLO-1	Explain key concepts of information security such as design principles, cryptography, risk management, and ethics	C2 (Explain)
CLO-2	Discuss legal, ethical, and professional issues in information security	A2 (Discuss)
CLO-3	Apply various security and risk management tools for achieving information security and privacy	C3 (Apply)
CLO-4	Identify appropriate techniques to tackle and solve problems in the discipline of information security	C4 (Identify)

Aims and Objectives	1. In this course students learn basics of information security, in both management aspect and technical aspect. 2. Students understand of various types of security incidents and attacks, and learn methods to prevent, detect and react incidents and attacks. Students will also learn basics of application of cryptography which are one of the key technologies to implement security functions.
Learning Outcomes	• CLO-1: Explain key concepts of information security such as design principles, cryptography, risk management, and ethics • CLO-2: Discuss legal, ethical, and professional issues in information security • CLO-3: Apply various security and risk management tools for achieving information security and privacy • CLO-4: Identify appropriate techniques to tackle and solve problems in the discipline of information security
Syllabus	I. Information security foundations, security design principles; security mechanisms, symmetric and asymmetric cryptography, encryption, hash functions, digital signatures, key management, authentication and access control; software security, vulnerabilities and protections, malware, database security; network security, firewalls, intrusion detection; security policies, policy formation and enforcement, risk assessment, cybercrime, law and ethics in information security, privacy and anonymity of data.
Contents	II. **Computer Security Concepts** i. Threats, Attacks, and Assets ii. Security Functional Requirements iii. Fundamental Security Design Principles iv. Attack Surfaces and Attack Trees v. Computer Security Strategy vi. Standards III. **Cryptographic Tools** i. Confidentiality with Symmetric Encryption ii. Message Authentication and Hash Functions iii. Public-Key Encryption iv. Digital Signatures and Key Management v. Random and Pseudorandom Numbers vi. Practical Application: Encryption of Stored Data

	IV. **User Authentication** i. Digital User Authentication Principles ii. Password-Based Authentication iii. Token-Based Authentication iv. Biometric Authentication v. Remote User Authentication vi. Security Issues for User Authentication vii. Practical Application: An Iris Biometric System V. **Access Control** i. Access Control Principles ii. Subjects, Objects, and Access Rights iii. Discretionary Access Control iv. Example: UNIX File Access Control v. Role-Based Access Control vi. Attribute-Based Access Control vii. Identity, Credential, and Access Management viii. Trust Frameworks VI. **Database and Data Centre Security** i. The Need for Database Security ii. Database Management Systems iii. Relational Databases iv. SQL Injection Attacks v. Database Access Control vi. Inference vii. Database Encryption viii. Data Center Security VII. **Malicious Software** i. Types of Malicious Software ii. Advanced Persistent Threat iii. Propagation — Infected Content - Viruses iv. Propagation — Vulnerability Exploit - Worms v. Propagation — Social Engineering — SPAM E-Mail, Trojans vi. Payload — System Corruption vii. Payload — Attack Agent — Zombie, Bots viii. Payload — Information Theft — Keyloggers, Phishing, Spyware ix. Payload — Stealthing — Backdoors, Rootkits x. Countermeasures

	VIII. **Denial-of-Service Attacks** i. Denial-of-Service Attacks ii. Flooding Attacks iii. Distributed Denial-of-Service Attacks iv. Application-Based Bandwidth Attacks v. Reflector and Amplifier Attacks vi. Defenses Against Denial-of-Service Attacks vii. Responding to a Denial-of-Service Attack IX. **Intrusion Detection** i. Intruders ii. Intrusion Detection iii. Analysis Approaches iv. Host-Based Intrusion Detection v. Network-Based Intrusion Detection vi. Distributed or Hybrid Intrusion Detection vii. Intrusion Detection Exchange Format viii. Honeypots X. **Firewalls and Intrusion Prevention Systems** i. The Need for Firewalls ii. Firewall Characteristics and Access Policy iii. Types of Firewalls iv. Firewall Basing v. Firewall Location and Configurations vi. Intrusion Prevention Systems XI. **IT Security Management and Risk Assessment** i. IT Security Management ii. Organizational Context and Security Policy iii. Security Risk Assessment iv. Detailed Security Risk Analysis XII. **Legal and Ethical Aspects** i. Cybercrime and Computer Crime ii. Intellectual Property iii. Privacy iv. Ethical Issues
Teaching-learning Strategies	The course will be based on the following teaching and learning activities:

<table>
<tr><td></td><td colspan="4">• Lectures covering the theoretical part using PowerPoint presentations
• Case studies
• Review questions</td></tr>
<tr><td>Assignments</td><td colspan="4">Total 4 Assignment</td></tr>
<tr><td rowspan="4">Assessment and Examinations</td><td>Sr. #</td><td>Elements</td><td>Weightage</td><td>Details</td></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>Assignments, Presentations, Quizzes.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>Mid Term exam of 90 Minutes in 9th week of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>End Term exam of 120-180 minutes at the end of semester.</td></tr>
<tr><td>Textbooks</td><td colspan="4">• Computer Security: Principles and Practice, 3rd edition by William Stallings</td></tr>
<tr><td>Reference Material/Suggested Readings</td><td colspan="4">• Whitman, M. E., & Mattord, H. J. (2019). Principles of information security.
• Gollmann, D. (2011. Computer security. Chichester: Wiley.
• Easttom, W., & Safari, an O'Reilly Media Company. (2011). Computer Security Fundamentals, Second Edition.
• Gordon, A. (2015). Official (ISC)2 Guide to the CISSP CBK, Fourth Edition. Hoboken: CRC Press.</td></tr>
<tr><td>Notes</td><td colspan="4">• Power Point slides with reading material from book.</td></tr>
</table>

Detailed Lecture wise plan

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
1	1	Computer Security Concepts Threats, Attacks, and Assets Security Functional Requirements	Ch-01	
	2	Fundamental Security Design Principles Attack Surfaces and Attack Trees Computer Security Strategy	Ch-01	
2	3	Cryptographic Tools Confidentiality with Symmetric Encryption Message Authentication and Hash Functions	Ch-02	
	4	Public-Key Encryption Digital Signatures and Key Management	Ch-02	
3	5	Random and Pseudorandom Numbers Practical Application: Encryption of Stored Data	Ch-02	
	6	User Authentication Electronic User Authentication Principles Password-Based Authentication	Ch-03	**Assignment-1**
4	7	Token-Based Authentication Biometric Authentication Remote User Authentication	Ch-03	**Quiz-1**
	8	Security Issues for User Authentication Practical Application: An Iris Biometric System	Ch-03	
5	9	Case Study: Security Problems for ATM Systems	Ch-03	

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
	10	Access Control Access Control Principles Subjects, Objects, and Access Rights Discretionary Access Control Example: UNIX File Access Control	Ch-04	
6	11	Role-Based Access Control Attribute-Based Access Control Identity, Credential, and Access Management Trust Frameworks	Ch-04	
	12	Case Study: RBAC System for a Bank	Ch-04	
7	13	Database and Cloud Security The Need for Database Security Database Management Systems	Ch-05	**Assignment-2**
	14	Relational Databases SQL Injection Attacks	Ch-05	**Quiz-2**
8	15	Database Access Control Inference Database Encryption	Ch-05	
	16	Cloud Computing Cloud Security Risks and Countermeasures	Ch-05	
9	17	Data Protection in the Cloud Cloud Security as a Service	Ch-05	
	18	Malicious Software Types of Malicious Software (Malware) Advanced Persistent Threat	Ch-06	
10	19	Propagation—Infected Content—Viruses Propagation—Vulnerability Exploit—Worms Propagation—Social Engineering—Spam E-Mail, Trojans	Ch-06	

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
	20	Payload—System Corruption Payload—Attack Agent—Zombie, Bots	Ch-06	
11	21	Payload—Information Theft—Keyloggers, Phishing, Spyware Payload—Stealthing—Backdoors, Rootkits Countermeasures	Ch-06	
	22	Denial-of-Service Attacks Flooding Attacks	Ch-07	
12	23	Distributed Denial-of-Service Attacks Application-Based Bandwidth Attacks Reflector and Amplifier Attacks	Ch-07	
	24	Defenses Against Denial-of-Service Attacks Responding to a Denial-of-Service Attack	Ch-07	
13	25	Intrusion Detection Intruders Intrusion Detection Analysis Approaches	Ch-08	**Assignment-3**
	26	Host-Based Intrusion Detection Network-Based Intrusion Detection Distributed or Hybrid Intrusion Detection	Ch-08	**Quiz-3**
14	27	Intrusion Detection Exchange Format Honeypots Intrusion Detection, Example System: Snort	Ch-08	
	28	Firewalls and Intrusion Prevention Systems The Need for Firewalls Firewall Characteristics and Access Policy	Ch-09	

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
15	29	Types of Firewalls Firewall Basing Firewall Location and Configurations Intrusion Prevention Systems Example: Unified Threat Management Products	Ch-09	**Assignment-4**
	30	IT Security Management and Risk Assessment IT Security Management Organizational Context and Security Policy Security Risk Assessment Detailed Security Risk Analysis	Ch-14	**Quiz-4**
16	31	Legal and Ethical Aspects Cybercrime and Computer Crime Security policies, Policy formation and enforcement	Ch-19	
	32	Cybercrime, law and ethics in information security, Privacy and anonymity of data. Intellectual Property Privacy Ethical Issues	Ch-19	

Program	BS Data Science	
Course Code	CC-110	
Course Title	Digital Logic Design	
Credit Hours	**Theory**	**Lab**
	3	1
Lecture Duration	90 minutes (1.5 Hours), 2 lectures per week, 3 hours lab session per week	
Semester	2	
Pre-requisites	**Courses**	**Knowledge**
Follow Up Courses	Computer Organization and Assembly Language	

Course Learning Outcomes (CLOs)		
CLO No	**Course Learning Outcome**	**Bloom Taxonomy**
CLO-1	Acquire knowledge related to the concepts, tools and - techniques for the design of digital electronic circuits	C2 (Understand)
CLO-2	Demonstrate the skills to design and analyze both combinational and sequential circuits using a variety of techniques.	C2 (Understand)
CLO-3	Apply the acquired knowledge to simulate and implement small-scale digital circuits	C3 (Apply)
CLO-4	Understand the relationship between abstract logic characterizations and practical electrical implementations.	C2 (Understand)

Aims and Objectives	1. Get the theoretical and the practical knowledge of the fundamental circuitry of the computers.
Learning Outcomes:	Student will have knowledge of • Number Systems and Boolean Algebra

	• Combinational Circuits • Sequential Circuits • Registers and Memory elements • Lab experience of ICS
Syllabus	Number Systems, Logic Gates, Boolean Algebra, Combination logic circuits and designs, Simplification Methods (K-Map, Quinn Mc-Cluskey method), Flip Flops and Latches, Asynchronous and Synchronous circuits, Counters, Shift Registers, Counters, Triggered devices & its types. Binary Arithmetic and Arithmetic Circuits, Memory Elements, State Machines. Introduction Programmable Logic Devices (CPLD, FPGA) Lab Assignments using tools such as Verilog HDL/VHDL, MultiSim
Contents	1. Number Systems, 2. Logic Gates, 3. Boolean Algebra, 4. Combination logic circuits and designs, 5. Simplification Methods (K-Map, Quinn Mc-Cluskey method), 6. Flip Flops and Latches, 7. Asynchronous and Synchronous circuits, 8. Counters, 9. Shift Registers, 10. Counters, Triggered devices & its types. 11. Binary Arithmetic and Arithmetic Circuits, 12. Memory Elements, 13. State Machines. 14. Introduction Programmable Logic Devices (CPLD, FPGA) Lab Assignments using tools such as Verilog HDL/VHDL, MultiSim
Teaching-learning Strategies	The students will be given all round knowledge of the subject oral/on white board/assignments/sudden quizzes during class rooms only.
Assignments	Assignments will be assigned throughout the course.

Assessment and Examinations

Sr. #	Elements	Weightage	Details
1	Formative Assessment	25%	It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior,

				hands-on-activities, short tests, quizzes etc.
	2	Midterm Assessment	35%	It takes place at the mid-point of the semester.
	3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.
Textbooks	1. Digital Fundamentals by Floyd, 11/e. 2. Fundamental of Digital Logic with Verilog Design, Stephen Brown, 2/e			
Reference Material/Suggested Readings	All reference Material/readings will be provided during lectures as per the class performance and their interest in the degree program overall.			
Notes	Students will take their own notes during class.			

<table>
<tr><td>Program</td><td colspan="3">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="3">CC-210</td></tr>
<tr><td>Course Title</td><td colspan="3">Computer Organization and Assembly Language Programming</td></tr>
<tr><td rowspan="2">Credit Hours</td><td>Theory</td><td colspan="2">Lab</td></tr>
<tr><td>3</td><td colspan="2">1</td></tr>
<tr><td>Lecture Duration</td><td colspan="3">90 minutes (1.5 Hours), 2 lectures per week, 3 hours lab session per week</td></tr>
<tr><td>Semester</td><td colspan="3">4</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td>Courses</td><td colspan="2">Knowledge</td></tr>
<tr><td>Digital Logic Design</td><td colspan="2">• Strong grip on Number system
• Strong back ground of Combinational and Sequential circuits</td></tr>
<tr><td>Follow Up Courses</td><td colspan="3">Computer Architecture</td></tr>
<tr><td colspan="4">Course Learning Outcomes (CLOs)</td></tr>
<tr><td>CLO No</td><td colspan="2">Course Learning Outcome</td><td>Bloom Taxonomy</td></tr>
<tr><td>CLO-1</td><td colspan="2">Acquire the basic knowledge of computer organization computer architecture and assembly language.</td><td>C2 (Understand)</td></tr>
<tr><td>CLO-2</td><td colspan="2">Understand the concepts of basic computer organization, architecture, and assembly language techniques</td><td>C2 (Understand)</td></tr>
<tr><td>CLO-3</td><td colspan="2">Solve the problems related to computer organization and assembly language</td><td>C3 (Apply)</td></tr>
<tr><td>Aims and Objectives</td><td colspan="3">1. Students will understand as to how a microprocessor is designed starting from a basic NAND gate to a full-blown computer system
2. Students will learn to write Hardware Description Language of various components of a computer system
3. Students will learn the different ways of interfacing I/O devices with computer
4. Students will learn to design the ISA, machine and assembly language of a microprocessor</td></tr>
</table>

	5. Students will understand the organization of the Intel processors, and be able to write basic programs for x86-64 Assembly Language
Learning Outcomes	• Students will understand, design and write the HDL of all the components of a Von-Neumann based computer. Understand the basic concept of computer organization, will design its assembly, machine language and will write its assembler in C, other than writing some basic assembly programs for that designed computer (Understand, Apply, Demonstrate) • Students will have a strong grip of x86-64 assembly language and the tool chain involved (Apply)
Syllabus	**Part-I**: The pre-mid part of the course deals with design of a complete computer system and writing its HDL. **Part-II**: The prost-mid part of the course deals with a detailed discussion on evolution of Intel processors, its programming models, its assembly language and programming tool chain
Contents	**<u>Section-1:</u>** - HDL for Combinational Circuits - HDL for Sequential Circuits - Data Storage in computer system **<u>Section-2:</u>** - Design and HDL of computer memory - Instruction Set Architecture - Design of hardware and writing HDL for Hack Computer - Interfacing I/O devices with Hack Computer **<u>Section-3:</u>** - Design of Machine Language of Hack Computer - Design of Assembly Language of Hack Computer - Design of Data Path (Buses) for Hack Computer - Design and Code of Hack Assembler **<u>Section-4:</u>** - History and Evolution of Intel Microprocessors - Concept of pipelining and improving processor performance - Programming Model of x86-64 processor - NASM and x86-64 assembly - Debugging with GNU debugger (gdb) **<u>Section-5:</u>** - Data transfer instructions - Memory addressing modes - X86-64 Logical and Bit shifting operations

<table>
<tr><td></td><td>- Control Transfer Instructions
- Function Calling Convention and Function Stack Frames

<u>Section-6:</u>
- Mixing C with Assembly programs
- Getting user input
- Programming with Arrays and Strings
- Floating Point Instructions
- Computer performance and parallel processing hardware
- Combinational & Sequential Circuits</td></tr>
<tr><td>Teaching-learning Strategies</td><td>• Lectures
• Case Studies
• Project
• Assignments</td></tr>
<tr><td>Assignments</td><td>Types and Number with calendar</td></tr>
<tr><td>Assessment and Examinations</td><td><table>
<tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.</td></tr>
</table></td></tr>
</table>

Textbooks	A. The Elements of Computing Systems, Building a modern computer, by Noam Nisan and Shimon Schocken, 2nd Ed, Published in 2020, ISBN-13: 978-0262640688 B. X86_64 Assembly Language Programming with Ubuntu, by Ed. Jorgensen, January 2020
Reference Material/Suggested Readings	C. Introduction to Computing Systems: from bits and gates to C and beyond, by Yale Patt and Sanjay Patel, 3rd Ed, Published in 2020, ISBN13: 9781260150537 D. X86_64 Assembly Language Programming with Ubuntu, by Ed. Jorgensen, January 2020 X86_64 Assembly Language Programming with Ubuntu, by Ed. Jorgensen, January 2020 E. Dr. Muhammad Arif Butt, COAL -Video Lectures: https://www.youtube.com/c/LearnWithArif/playlists
Notes	

Detailed Lecture wise plan

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
1	1	**Introduction to Course:** Computer Organization and Assembly Language Programming and a discussion on the course matrix. The course will be having two parts in the first half of the course we will be designing and writing the HDL of a full-blown computer, will design its machine and assembly language, will write programs and execute those programs on the designed h/w architecture. At the end of the first half of the course, we will also design and write an Assembler (in C) for the designed computer. The second half of the course will deal with the assembly of the all-time famous x86-64 architecture. The links from where the e-books, tools, code snippets, and lecture slides, and other misc resources can be downloaded are mentioned.	Text A-Ch1	
	2	**HDL for Combinational Circuits – I:** Review of Boolean logic and gates. Introduction to Hardware Description Languages. Design and code of And, Or, Not gates using the universal NAND gate. Downloading and installing Hardware Simulator and interactive chip testing on this simulator. Designing Xor chip and performing script based testing of Xor chip on h/w simulator. **HDL for Combinational Circuits – II:** Design and HDL code for XOR chip, using And, OR, Not gate chips. A demo of Verification of XOR chip using interactive chip testing in the h/w simulator. A brief overview of script-based chip testing. Writing script for testing of the designed XOR chip. A discussion on key players involved in a hardware construction project.	Text A-Ch1	**Lab:**

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
2	3	**HDL for Combinational Circuits – III:** Design and HDL code for some standard combinational circuits like Encoder, Decoders, Multiplexers, and De-Multiplexer chips. A demo of Verification / Testing of these standard combinational chips using interactive chip testing in the h/w simulator. **HDL for Combinational Circuits – IV:** Design and HDL code for multi-bit gates. The concept of buses and the design of chips having buses as input. Design and code of And16, Or16, Not16, and Mux16 chips having 16-bits inputs. Design and code of And4way16 and similar chips having four inputs with each input of 16 bits	Text A-Ch1	
	4	**Data Storage – I:** Data Representation in Computers, Unsigned, and Signed Numbers, Sign magnitude representation and its limitations, 1s Complement representation and its limitations, 2s Complement representation, Comparisons and pros and cons of each, Ranges and different Storage Sizes, Overflow in Unsigned and Signed Numbers, How the Hardware Detect an Overflow, Concept of Sign Extension, Encoding Characters and Strings (ASCII and Unicode)	Ref C-Ch2	**Lab:**
3	5	**Data Storage – II:** Encoding Real Numbers, Fixed Point Representation, Floating Point Representations (IEEE-754), Storage layout, Conversion Examples, Range and Precision, Arithmetic Operations, Overflow and Underflow, IEEE-754 Special Values	Ref C-Ch2	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
	6	**Design of ALU - I:** Review of HDL for Combinational Circuits, Designing a single bit Logic Unit Writing HDL for Combinational Arithmetic Circuits like Half Adder, Full Adder, Full Subtractor, 16-bit Binary Adder (Add16 chip), 16-bit Incrementer (Inc16 chip), Demo of above chips on H/W Simulator **Design of ALU - II:** Components of a Computer System, Design of ALU, The Hack ALU, The Hack ALU Operations, Design of Hack ALU, HDL of Hack ALU, Verifying the ALU chip on H/W Simulator	Text A-Ch2	**Lab:**
4	7	**Design of Sequential Circuits:** Why Sequential Circuits? Understanding Time in Circuits, Combinational vs Sequential Circuits, Flip Flops, D flip Flop, SR Flip Flop, JK Flip Flop, T Flip Flop **Design of Registers:** What are Registers, Design of 1-bit Register, HDL for 1-bit Register, Design of 16-bit Register, HDL for 16-bit Register	Text A Ch3	
	8	**Design of Memory:** Concept of Memory Hierarchy, Multi-Byte Read/Write, Design of Random Access Memory, Read/Write Logic of RAM, API of a RAM Chip, HDL of 8 Words RAM, HDL of 64 Words RAM, HDL of 512 Words RAM, HDL of 4K Words RAM, HDL of 16K Words RAM **Design of Counters:** Overview of Hack Computer Components, Overview of Counters, Why do we need Counter for our Hack Computer, Concept of Program Counter, Counter Simulation, Design and Implementation of PC for Hack Computer, Demo on H/W Simulator	Text A-Ch3	**Lab:**
5	9	**ISA-I:** Overview of Computer System, Universality of Computer System, Turing Machine, Von Neumann Architecture, Instruction Set Architecture (ISA)	Ref A Ch3	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
	10	**ISA-II:** Five Dimensions of ISA, Class of ISA, Types and Sizes of Operands, Operations (including control flow instructions), Memory Addressing Models and Addressing Modes, Encoding an ISA	Text A-Ch3	
6	11	**Hack Machine Language – I:** Hack Computer Machine Language, Review of h/w of Hack Computer, Software of Hack Computer, A Instruction, C Instruction, Examples	Text A Ch4	
	12	**Hack Machine Language – II:** Review of Hack Symbolic Machine Instructions, A Instruction, C Instruction, Binary Code Format of Hack Computer Instruction, Encoding of 16 bit A-Instruction, Encoding of 16 bit C-Instruction, Examples, A Complete Hack Program: Assembly Language	Text A-Ch4	
7	13	**Interfacing I/O Devices:** How to interface I/O devices with computer, Interfacing Screen with Hack computer, Demo of built-in Screen chip on h/w Simulator, Interfacing Keyboard with Hack computer, Demo of built-in Keyboard chip on h/w Simulator	Text A Ch 5	
	14	**Hack Assembly Programming – I:** Review of Hack Computer Assembly Instructions, Hack Assembly Programs, A Hello World in Hack assembly, CPU Emulator, Demo, Program Termination **Hack Assembly Programming – II:** Recap previous lecture, Symbols in Hack Assembly Language, Built-in Symbols, Label Symbols, Variable Symbols, Branching, Iteration	Text A-Ch4	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
8	15	**Hack Assembly Programming – III:** Review of Hack Assembly Programs, Pointers and Arrays, Input / Output Instructions, Debugging, Review of Hack Assembly Programs, Pointers and Arrays, Input / Output Instructions, Debugging	Text A-Ch4	
	16	**Data path of Hack CPU-I:** Von Neumann Architecture, Flow of Information inside Computers, Buses, Data Bus, Address Bus, Control Bus, Fetch Execute Cycle, Fetch Execute Clash, Harvard Architecture **Data path of Hack CPU-II:** Review of Hack Computer Architecture, Hack CPU Interface, Hack CPU Implementation, Input/output and Operations of Hack ALU, Control Logic of Hack CPU	Text A-Ch5	
9	17	**Design of Hack Computer:** Recap of Hack Computer Architecture, Implementation of Hack CPU Chip (CPU.hdl), Implementation of Hack Memory Chip (Memory.hdl), RAM16 chip (RAM16K.hdl), Screen chip (Screen.hdl), Keyboard chip (Keyboard.hdl), Implementation of Hack ROM Chip (ROM32K.hdl), Implementation of Hack Computer Chip (Computer.hdl)	Text A-Ch5	
	18	**Design of Hack Assembler:** What is an Assembler? How an Assembler works? Hack Machine Language Specification, Demo of Built-in Hack Assembler, Design of Hack Assembler (w/o Symbols), Design of Hack Assembler (with Symbols), Hack Assembler Implementation in C/C , Executing Hack Machine Code, Hack Computer Chip in h/w Simulator, CPU Emulator	Text A-Ch6	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
10	19	**History and Evolution of Intel Microprocessors:** Intel 4004 (1971), Intel 8008, Intel 8080, Intel 8086 (x86), Intel 80286, Intel 80386, Intel 80486, Intel 80586 (Pentium P5), Intel 80686 (Pentium P6), Intel Core (2006) Intel Nehalam (2008), Intel Sandy Bridge, Intel Ivy Bridge, Intel Haswell, Intel Broadwell, Intel Sky Lake, Intel Kaby Lake, Intel Coffee Lake, Intel Coffee Lake Refresh, Intel Comet Lake (2019) **On Improving Processors Performance:** CPU Performance Equation, Single Cycle vs Multi Cycle CPU Architecture, Pipelined CPU Architecture, Pipeline Stages, Even vs Uneven pipelined stages, Pipelined Hazards, Solutions of Pipeline Hazards, CISC vs RISC Architecture	Text B-Ch1	
	20	**Programming Model of x86 Architecture:** Layout of memory models (flat, segmented) and register set file of Intel 8080, 80386, x86-64. Logical to physical address translation for segmented memory model.	Text B-Ch2	
11	21	**Hello World in x86-64 Assembly:** Overview of microprocessor families and their corresponding assembly languages. Tool chain and programming environment for x86-64 assembly programming. Running the first hello world assembly program. **Structure of x86-64 Assembly Program:** A discussion on the x86-64 assembly language instruction format and the overall structure of assembly	Text B-Ch4	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
	22	**Debugging C-Program with gdb:** A review of C-compilation process. What is a debugger? Why use gdb? How to compile, load and run a program inside gdb and get information about the running process. Getting help inside gdb, setting break points, watch points, stepping through the code, examining and modifying variables, convenience variables and setting conditional break points. **Data Types and Endianness in x86-64:** Usage of different data types and special tokens in NASM. Practically understand about the endianness of a machine	Text B-Ch5	
12	23	**Data Transfer Instructions and Process Stack:** Usage of different move instructions like mov, movzx, movsx, lea and xchg. A discussion on the working of process stack and the push and pop instructions. **Memory Addressing Modes:** Theoretical concepts and pros and cons of addressing modes used by different processors. Addressing modes used by x86-64 like Base-Index-Sale-Displacement	Text B-Ch8	
	24	**Arithmetic Instructions Part-I:** A recap of x86-64 register set and the programming tool chain. Summary of major categories of x86-64 instructions. A practical demo on the use of add, adc, sub, sbb, inc, dec, neg, cmp, clc, stc, and cmc. A discussion on how the flags are effected after these arithmetic instructions **Arithmetic Instructions Part-II:** A recap of x86-64 register set and the programming tool chain. Summary of major categories off x86-64 instructions. A practical demo on the use of mul, div, imul, idiv instructions. A discussion on how the flags are effected after these arithmetic instructions	Text B-Ch7	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
13	25	**Logical Operations:** A recap of x86-64 register set and the programming tool chain. Summary of major categories of x86-64 instructions. A practical demo on the use of and, or, not, xor, and test instructions. A discussion on how the flags are effected after these logical instructions. **Bit-Shifting Operations:** A recap of x86-64 register set and the programming tool chain. Summary of major categories of x86-64 instructions. A practical demo on the use of shl, sal, shr, sar, rol, ror, rcl, and rcr instructions. A discussion on how the flags are effected after these logical instructions.	Text B-Ch7	
	26	**Control Instructions - I:** A discussion on control of flow of execution of a program and how to change it. Description of unconditional jump instruction with a demonstration of example programs. Discussion on signed and unsigned conditional jump instructions with demonstration of example programs. Translating if…else code too assembly language. **Control Instructions - II:** A Recap of previous session. Translating high level repetition structure (for, while, ….) to its corresponding x86 assembly code using conditional jump instructions as well as using x86 loop instructions.	Text B-Ch7	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
14	27	**GDB with PEDA Plugin:** A recap of gdb command line and text user interface mode. Downloading, installing, and configuring Python Exploit Development Assistance (PEDA) plugin to enhance the firepower of gdb. Debugging the x86-64 assembly programs using gdb with PEDA and a brief intro of using this plugin for reverse engineering and exploit development **Functions in Assembly Language – I:** What are functions? Why they are used in programming languages? Syntax of defining an assembly function in NASM and MASM. Understanding the working of x86-64 call and ret instruction. The usage of process run-time stack in function call. The concept and requirement of caller-saved and callee-saved registers	Text B-Ch9,12	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
	28	**Functions in Assembly Language – II:** Recap of functions in assembly language and use of call and ret instructions. How different operating systems allow x86-64 assembly programmers to pass and return values to and from functions. Designing an assembly function to display a single digit and a multi-digit decimal number on screen. Writing, assembling, linking and executing multi-file assembly programs on x86-64. **Function Calling Convention and FSF:** Recap of assembly language functions. Understanding function calling in high-level languages like C and C . The concept of Function Stack Frame (FSF) or Activation Record used to store data associated with a high-level function on the process run time stack. The x86-64 procedure prolog and procedure epilog for creating and removing FFSF from the stack. A demonstration of stack-based buffer overflow vulnerability and concept of exploiting it.	Text B-Ch9,12	
15	29	**Mixing C with x86-64 Assembly:** Recap of pushing and popping FSF to and from the process run time stack in high level languages. Calling C-Library functions from within an assembly program. Doing the reverse, i.e., Calling assembly functions from within a C program. **Getting User Input:** User input via system calls, library calls, and command line arguments. What are command line arguments and why we use them in high level languages as well as in assembly programming? Converting the string input received via command line to integer for further processing.	Text B-Ch13,16	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
	30	**Arrays:** Array address computation (pointer arithmetic) based on its types (bytes, words, double words and quad-words), General pattern for memory references, allocation arrays using malloc, processing arrays, filling with random numbers, printing array elements **String:** Understanding C-strings in assembly, x86 string instructions to store strings in memory, load strings from memory, comparing strings, and scanning strings for substrings	Text B-Ch13,16	
16	31	**Floating Point Instructions:** 8087 floating point instructions that use stack of 80 bit floating point registers (ST0, ST1, …). Intel Core iseries floating point instructions that work with Streaming SIMD Extensions (SSE) 128-bit registers (xmm0, xmm1, …). The concept of Advanced Vector Extensions (AVX) that use 256-bit registers (ymm0, ymm1, …) Data movement instructions (movss, movsd) Arithmetic instructions (addss, addsd, subss, subsd, mulss, mulsd, divss, divsd) Integer / floating point conversion instructions (cvtss2sd, cvtsd2ss, cvtss2si, cvtsi2ss, cvtsd2si, cvtsi2sd) Floating point control instructions (ucomiss, ucomisd) Floating point calling convention	Text B-Ch18	
	32	**Computer Performance and Parallel Processing Hardware:.**	Text B-Ch19	

<table>
<tr><td>Program</td><td colspan="3">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="3">CC-310</td></tr>
<tr><td>Course Title</td><td colspan="3">Artificial Intelligence</td></tr>
<tr><td rowspan="2">Credit Hours</td><th>Theory</th><th colspan="2">Lab</th></tr>
<tr><td>3</td><td colspan="2">1</td></tr>
<tr><td>Lecture Duration</td><td colspan="3">90 minutes (1.5 Hours), 2 lectures per week, 3 hours lab session per week</td></tr>
<tr><td>Semester</td><td colspan="3">4</td></tr>
<tr><td rowspan="2">Pre-requisites</td><th>Courses</th><th colspan="2">Knowledge</th></tr>
<tr><td>Nil</td><td colspan="2">Basic knowledge of Programming and Data Structures would be helpful</td></tr>
<tr><td>Follow Up Courses</td><td colspan="3">Introduction to Data Science</td></tr>
<tr><td colspan="4">Course Learning Outcomes (CLOs)</td></tr>
<tr><td>CLO No</td><td colspan="2">Course Learning Outcome</td><td>Bloom Taxonomy</td></tr>
<tr><td>CLO-1</td><td colspan="2">Understand the fundamental constructs of Python programming language.</td><td>C2 (Understand)</td></tr>
<tr><td>CLO-2</td><td colspan="2">Understand key concepts in the field of artificial intelligence</td><td>C2 (Understand)</td></tr>
<tr><td>CLO-3</td><td colspan="2">Implement artificial intelligence techniques and case studies</td><td>C3 (Apply)</td></tr>
<tr><td>Aims and Objectives</td><td colspan="3">1. This course aims to introduce students to the exciting and diverse field of Artificial Intelligence (AI).
2. To provide coverage of fundamental concepts of symbolic manipulations, pattern matching, knowledge representation, and decision making.</td></tr>
</table>

	3. The objective of this course is to equip students with the basic problem-solving techniques used in AI so that they are able to apply that knowledge to the real-world problems.
Learning Outcomes	• Understanding key concepts in the field of AI • Understanding the fundamental constructs of AI programming languages, e.g. Prolog, LISP etc. • Implementing AI techniques to solve real-world problems
Syllabus	An Introduction to Artificial Intelligence and its applications towards Knowledge Based Systems; Introduction to Reasoning and Knowledge Representation, Problem Solving by Searching (Informed searching, Uninformed searching, Heuristics, Local searching, Min-max algorithm, Alpha beta pruning, Game-playing); Case Studies: General Problem Solver, Eliza, Student, Macsyma; Learning from examples; Natural Language Processing; Recent trends in AI and applications of AI algorithms. Lisp & Prolog programming languages will be used to explore and illustrate various issues and techniques in Artificial Intelligence.
Contents	Unit 1: Introduction 1.1 Discussion on the concepts of Intelligence and AI 1.2 History of AI 1.3 Strong Vs Weak AI, Strong Vs Weak method problems solving 1.4 Reasoning and knowledge representation 1.5 Physical Symbol System Hypothesis Unit 2: Problem Solving by Searching 2.1 Uninformed search

	2.2 Informed search 2.3 Local search heuristics 2.4 Game playing: Minimax algorithm, alpha-beta pruning Unit 3: Reasoning in AI systems 3.1 Introduction to logic and reasoning in AI 3.2 Recap of Propositional and Predicate Calculi 3.3 Representation in formal logic 3.4 Automated reasoning 3.5 Resolution theorem proving Unit 4: Knowledge Based Systems: 4.1 Various types of knowledge-based systems (KBS) 4.2 Architecture of rule based Expert Systems 4.3 Case Studies: General Problem Solver, Eliza, Student etc. Unit 5: Natural Language Processing 5.1 Introduction 5.2 Phases of linguistic analysis 5.3 NLP system Unit 6: Learning in AI systems: 6.1 Genetic Models of learning 6.2 Symbolic vs Connectionist learning in AI 6.3 Artificial Neural networks: 6.3.1 Perceptrons 6.3.2 Multilayer Perceptrons 6.3.3 Deep Neural Networks

<table>
<tr><td>Teaching-learning Strategies</td><td colspan="4">Multimedia presentations involving interaction from students
Hands on exercises for concept reinforcement
Coding in laboratory</td></tr>
<tr><td>Assignments</td><td colspan="4">There would be 4 to 5 programming assignments (2 pre and 2-3 post midterm)</td></tr>
<tr><td rowspan="4">Assessment and Examinations</td><td>Sr. #</td><td>Elements</td><td>Weightage</td><td>Details</td></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.</td></tr>
<tr><td>Textbooks</td><td colspan="4">Luger, G. F. (2009). Artificial Intelligence- Structures & Strategies for Complex Problem Solving. (6th Edition). Pearson Education, Inc.
ISBN-13: 978-0-321-54589-3</td></tr>
</table>

Reference Material/Suggested Readings	• Russell, S., Norvig, P. (2015). Artificial Intelligence. A Modern Approach (3rd Edition). Pearson Education, Inc. ISBN-13: 978-0136042594 • Norvig, P. (1992). Paradigms of Artificial Intelligence Programming: Case studies in Common Lisp. Morgan Kaufman Publishers, Inc. ISBN-13: 978-1558601918 • Bratko, I. (2011). Prolog: Programming for Artificial Intelligence. (4th Edition). Pearson Education, Canada. ISBN-13: 978-0321417466
Notes	• Academic integrity is expected of all students. Plagiarism or cheating in any assessment will result in at least an F grade in the course, and possibly more severe penalties • There is no makeup for a missed sessional grading instruments like quizzes, assignments, and homework • The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester

Detailed Lecture wise plan

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
1	1	Introduction: definition, concept of intelligence; attributes of intelligence; History of AI	Ch#1	
	2	Schools of thought, methods of problem solving; reasoning and representation; physical symbol system hypothesis		Reading-1
2	3	Problem solving via search: uniformed search	Ch#3	
	4	Heuristic search techniques	Ch#4	
3	5	Properties of heuristics		Quiz-1
	6	Heuristics in game playing; minimax algorithm; alpha-beta technique		
4	7	Reasoning in AI systems: introduction to logical reasoning; recap of propositional and predicate calculus	Ch#2	Assignment-1; Reading-2
	8	Representation in formal logic; unification algorithm		
5	9	Automated reasoning; resolution theorem proving	Ch#14	
	10	Examples of resolution theorem proving		
6	11	Introduction to logic programming. Horn clauses		Quiz-2
	12	Prolog as an example logic programming system	Handouts	
7	13	Knowledge based systems: types, architecture of rule-based expert systems	Ch#8	Reading-3
	14	Expert system shells		Assignment-2
8	15	Case studies: GPS, Eliza	Handouts	
	16	Midterm review		
Midterm Exam				
9	17	Natural Language Processing: Introduction; phases of linguistic analysis	Ch#15	Reading-4
	18	NLP system overview		
10	19	Genetic Models of learning: Introduction; Genetic Algorithm (GA)	Ch#12	Reading-5
	20	Representation, fitness function, selection techniques		

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
11	21	Genetic operators, examples and implementation of GA		Quiz-3
	22	Symbolic VS connectionist learning;	Ch#10	Assignment-3
12	23	Naïve Bayes classification	Handouts	
	24	Decision Trees: ID3 algorithm; C 4.5 algorithm	Handouts	Reading-6
13	25	Inductive bias of decision tree learning; examples and implementation		
	26	Connectionist paradigm of learning: Neuron and Neural Networks; Artificial Neural Networks	Ch#11	
14	27	Perceptron networks; delta rule; linear separability problem		Reading-7
	28	Multilayer perceptrons; generalized delta rule		
15	29	Backpropagation algorithm		Assignment-4
	30	Issues and enhancements of backpropagation algorithm	Handouts	Quiz-4
16	31	Deep neural networks	Handouts	
	32	Final term review		
Final Exam				

Program	BS DS	
Course Code	CC-313	
Course Title	Analysis of Algorithms	
Credit Hours	**Theory**	**Lab**
	3	0
Lecture Duration	90 minutes (1.5 Hours), 2 lectures per week	
Semester	5	
Pre-requisites	**Courses**	**Knowledge**
	Data Structures and Algorithms	Nil
Follow Up Courses	Nil	

Course Learning Outcomes (CLOs)

CLO No	**Course Learning Outcome**	**Bloom Taxonomy**
CLO-1	Explain what is meant by “best”, “expected”, and “worst” case behavior of an algorithm	C2
CLO-2	Identify the characteristics of data and/or other conditions or assumptions that lead to different behaviors.	C3
CLO-3	Determine informally the time and space complexity of simple algorithms	C3
CLO-4	List and contrast standard complexity classes	C4
CLO-5	Use big O, Omega, Theta notation formally to give asymptotic upper bounds on time and space complexity of algorithms	C3
CLO-6	Use of the strategies(brute-force, greedy, divide-and- conquer, and dynamic programming) to solve an appropriate problem	C3
CLO-7	Solve problems using graph algorithms, including single- source and all-pairs shortest paths, and at least one minimum spanning tree algorithm	C3
CLO-8	Trace and/or implement a string-matching algorithm	C3

Aims and Objectives	1. Students can explain what is meant by “best”, “expected”, and “worst” case behavior of an algorithm 2. Students can determine informally the time and space complexity of simple algorithms. 3. Students can use of the strategies (brute-force, greedy, divide-and-conquer, and dynamic programming) to solve an appropriate problem 4. Students can solve problems using graph algorithms, including single-source and all-pairs shortest paths, and at least one minimum spanning tree algorithm
Learning Outcomes	• Students can explain what is meant by “best”, “expected”, and “worst” case behavior of an algorithm • Students can determine informally the time and space complexity of simple algorithms. • Students can use of the strategies (brute-force, greedy, divide-and-conquer, and dynamic programming) to solve an appropriate problem Students can solve problems using graph algorithms, including single-source and all-pairs shortest paths, and at least one minimum spanning tree algorithm • Students can trace and/or implement a string-matching algorithm
Syllabus	Introduction; Role of algorithms in computing, Analysis on nature of input and size of input Asymptotic notations; Big-O, Big Ω, Big Θ, little-o, little-ω, Sorting Algorithm analysis, loop invariants, Recursion and recurrence relations; Algorithm Design Techniques, Brute Force Approach, Divide-and-conquer approach; Merge, Quick Sort, Greedy approach; Dynamic programming; Elements of Dynamic Programming, Search trees; Heaps;

	Hashing; Graph algorithms, shortest paths, sparse graphs, String matching; Introduction to complexity classes
Contents	1. Introduction 2. Role of algorithms in computing 3. Analysis on nature of input and size of input Asymptotic notations; Big-O, Big Ω, Big Θ, little-o, little-ω 4. Sorting Algorithm analysis 5. Loop invariants 6. Recursion and recurrence relations 7. Algorithm Design Techniques 7.1. Brute Force Approach, 7.2. Divide-and-conquer approach 7.2.1. Merge, Quick Sort, 7.3. Greedy approach 7.4. Dynamic programming 7.4.1. Elements of Dynamic Programming 8. Search trees 9. Heaps 10. Hashing 11. Graph algorithms 11.1. shortest paths, 11.2. sparse graphs, 12. String matching 13. Introduction to complexity classes
Teaching-learning Strategies	• Interactive class session • Hands on practices in class • Brainstorming and Group discussion sessions
Assignments	• Paper based written assignments 6

<table>
<tr><td rowspan="4">Assessment and Examinations</td><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.</td></tr>
<tr><td>Textbooks</td><td colspan="4">Introduction to Algorithms (3rd edition) by Thomas H. Corman, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein
Algorithm Design, (1st edition, 2013/2014), Jon Kleinberg, Eva Tardos
Algorithms, (4th edition, 2011), Robert Sedgewick, Kevin Wayne</td></tr>
<tr><td>Reference Material/Suggested Readings</td><td colspan="4">Handout provided by the teacher.
PowerPoint Presentations
Various books Chapters / Notes
Internet resources</td></tr>
</table>

Notes	• Academic integrity is expected of all students. Plagiarism or cheating in any assessment will result in at least an F grade in the course, and possibly more severe penalties. • You bear all the responsibility for protecting your assignments from plagiarism. If anyone else submits your assignment or uses your code in his/her assignment, you will be considered equally responsible. • The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester. • There is no makeup for a missed sessional grading instruments like quizzes, assignments, and homework's.

Program	BS Data Science	
Course Code	DD-409	
Course Title	Parallel and Distributed Computing	
Credit Hours	**Theory**	**Lab**
	2	1
Lecture Duration	60 minutes (1 Hours), 2 lectures per week, 3 hours lab session per week	
Semester	7	
Pre-requisites	**Courses**	**Knowledge**
	Data Structures and Algorithms, Operating Systems	
Follow Up Courses		

Course Learning Outcomes (CLOs)

CLO No	Course Learning Outcome	Bloom Taxonomy
CLO-1	Learn about parallel and distributed computers.	C2 (Understand)
CLO-2	Write portable programs for parallel or distributed architectures using Message-Passing Interface (MPI) library	C2 (Understand)
CLO-3	Analyze complex problems with shared memory programming with openMP.	C4 (Analyze)

Aims and Objectives	1. Students will learn the parallel execution of the code parts. They will design the logic of the code so they parallel execution is possible.
Learning Outcomes:	After the theory and lab, the student and code the programs to exploit parallel architecture of the multicore computers and distributed computing environment.

<table>
<tr><td>Syllabus</td><td>Asynchronous/synchronous computation/communication, concurrency control, fault tolerance, GPU architecture and programming, heterogeneity, interconnection topologies, load balancing, memory consistency model, memory hierarchies, Message passing interface (MPI), MIMD/SIMD, multithreaded programming, parallel algorithms & architectures, parallel I/O, performance analysis and tuning, power, programming models (data parallel, task parallel, process-centric, shared/distributed memory), scalability and performance studies, scheduling, storage systems, synchronization, and tools (Cuda, Swift, Globus, Condor, Amazon AWS, OpenStack, Cilk, gdb, threads, MPICH, OpenMP, Hadoop, FUSE).</td></tr>
<tr><td>Contents</td><td>1. Asynchronous/synchronous computation/communication,
2. Concurrency control, fault tolerance,
3. GPU architecture and programming,
4. Heterogeneity, interconnection topologies, load balancing, memory consistency model,
5. Memory hierarchies,
6. Message passing interface (MPI), MIMD/SIMD, multithreaded programming, parallel algorithms & architectures, parallel I/O
7. Performance analysis and tuning, power, programming models (data parallel, task parallel, process-centric, shared/distributed memory), scalability and performance studies, scheduling, storage systems,
8. Synchronization, and tools (Cuda, Swift, Globus, Condor, Amazon AWS, OpenStack, Cilk, gdb, threads, MPICH, OpenMP, Hadoop, FUSE).</td></tr>
<tr><td>Teaching-learning Strategies</td><td>The students will be given all round knowledge of the subject oral/on white board/assignments/sudden quizzes during class rooms only.</td></tr>
<tr><td>Assignments</td><td>Assignments will be assigned throughout the course.</td></tr>
<tr><td>Assessment and Examinations</td><td><table>
<tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
</table></td></tr>
</table>

	2	Midterm Assessment	35%	It takes place at the mid-point of the semester.
	3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.
Textbooks	• Distributed Systems: Principles and Paradigms, A. S. Tanenbaum and M. V. Steen, Prentice Hall, 2nd Edition, 2007 • Distributed and Cloud Computing: Clusters, Grids, Clouds, and the Future Internet, K Hwang, J Dongarra and GC. C. Fox, Elsevier, 1st Ed.			
Reference Material/Suggested Readings	All reference Material/readings will be provided during lectures as per the class performance and their interest in the degree program overall.			
Notes	Students will take their own notes during class.			

<table>
<tr><td>Program</td><td colspan="3">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="3">DD-223</td></tr>
<tr><td>Course Title</td><td colspan="3">Advance Statistics</td></tr>
<tr><td rowspan="2">Credit Hours</td><td colspan="2">Theory</td><td>Lab</td></tr>
<tr><td colspan="2">3</td><td>0</td></tr>
<tr><td>Lecture Duration</td><td colspan="3">90 minutes (1.5 Hours), 2 lectures per week</td></tr>
<tr><td>Semester</td><td colspan="3">4</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td colspan="2">Courses</td><td>Knowledge</td></tr>
<tr><td colspan="2">Probability and Statistics</td><td></td></tr>
<tr><td>Follow Up Courses</td><td colspan="3"></td></tr>
<tr><td colspan="4">Course Learning Outcomes (CLOs)</td></tr>
<tr><td>CLO No</td><td colspan="2">Course Learning Outcome</td><td>Bloom Taxonomy</td></tr>
<tr><td>CLO-1</td><td colspan="2">Apply preprocessing techniques on any given raw data.</td><td>C3 (Apply)</td></tr>
<tr><td>CLO-2</td><td colspan="2">Select and apply proper data mining algorithm to discover interesting patterns.</td><td>C3 (Apply)</td></tr>
<tr><td>CLO-3</td><td colspan="2">Analyze and extract patterns to solve problems and point out how to deploy solution</td><td>C4 (Analyze)</td></tr>
<tr><td>CLO-4</td><td colspan="2">Evaluate systematically supervised, semi supervised and unsupervised models and algorithms with respect to their accuracy</td><td>C4 (Analyze)</td></tr>
<tr><td>Aims and Objectives</td><td colspan="3">○ Students will learn advanced statistical techniques to analyze the data to the next level.</td></tr>
<tr><td>Learning Outcomes:</td><td colspan="3">Students will become sufficiently confident to analyze data</td></tr>
<tr><td>Syllabus</td><td colspan="3">Introduction to Statistics, Use of Statistics in Data Science, Experimental Design, Statistical Techniques for Forecasting, Interpolation/ Extrapolation,</td></tr>
</table>

	Introduction to Probability, Conditional Probability, Prior and Posterior Probability, Random number generation (RNG), Techniques for RNG, Correlation analysis, Chi Square Dependency tests, Diversity Index, Data Distributions Multivariate Distributions, Error estimation, Confidence Intervals, Linear transformations, Gradient Descent and Coordinate Descent, Likelihood inference, Revision of linear regression and likelihood inference, Fitting algorithms for nonlinear models and related diagnostics, Generalized linear model; exponential families; variance and link functions, Proportion and binary responses; logistic regression, Count data and Poisson responses; log-linear models, Overdispersion and quasi-likelihood; estimating functions, Mixed models, random effects, generalized additive models and penalized regression; Introduction to SPSS, Probability/ Correlation analysis/ Dependency tests/ Regression in SPSS.
Contents	1. Introduction to Statistics, Use of Statistics in Data Science, 2. Experimental Design, Statistical Techniques for Forecasting, 3. Interpolation/ Extrapolation, Introduction to Probability, Conditional Probability, Prior and Posterior Probability, 4. Random number generation (RNG), Techniques for RNG, Correlation analysis, 5. Chi Square Dependency tests, Diversity Index, Data Distributions Multivariate Distributions, Error estimation, Confidence Intervals, 6. Linear transformations, Gradient Descent and Coordinate Descent, Likelihood inference, Revision of linear regression and likelihood inference, 7. Fitting algorithms for nonlinear models and related diagnostics, Generalized linear model; exponential families; variance and link functions, 8. Proportion and binary responses; logistic regression, Count data and Poisson responses; log-linear models, Overdispersion and quasi-likelihood; estimating functions, 9. Mixed models, random effects, generalized additive models and penalized regression; 10. Introduction to SPSS, Probability/ Correlation analysis/ Dependency tests/ Regression in SPSS.
Teaching-learning Strategies	The students will be given all round knowledge of the subject oral/on white board/assignments/sudden quizzes during class rooms only.
Assignments	Assignments will be assigned throughout the course.
	Sr. # | Elements | Weightage | Details

Assessment and Examinations	1	Formative Assessment	25%	It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.
	2	Midterm Assessment	35%	It takes place at the mid-point of the semester.
	3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.
Textbooks	1. Probability and Statistics for Computer Scientists, 2nd Edition, Michael Baron. 2. Probability for Computer Scientists, online Edition, David Forsyth 3. Discovering Statistics using SPSS for Windows, Andy Field			
Reference Material/Suggested Readings	All reference Material/readings will be provided during lectures as per the class performance and their interest in the degree program overall.			
Notes	Students will take their own notes during class.			

<table>
<tr><td>Program</td><td colspan="3">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="3">DD-221</td></tr>
<tr><td>Course Title</td><td colspan="3">Introduction to Data Science</td></tr>
<tr><td rowspan="2">Credit Hours</td><td colspan="2">Theory</td><td>Lab</td></tr>
<tr><td colspan="2">2</td><td>1</td></tr>
<tr><td>Lecture Duration</td><td colspan="3">90 minutes (1 Hours), 2 lectures per week, 3 hours lab session per week</td></tr>
<tr><td>Semester</td><td colspan="3">5</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td colspan="2">Courses</td><td>Knowledge</td></tr>
<tr><td colspan="2"></td><td>Nil</td></tr>
<tr><td>Follow Up Courses</td><td colspan="3">Data Mining, Data Warehousing and Business Intelligence</td></tr>
<tr><td colspan="4">Course Learning Outcomes (CLOs)</td></tr>
<tr><td>CLO No</td><td colspan="2">Course Learning Outcome</td><td>Bloom Taxonomy</td></tr>
<tr><td>CLO-1</td><td colspan="2">Describe what Data Science is and the skill sets needed to be a data scientist.</td><td>C2 (Understand)</td></tr>
<tr><td>CLO-2</td><td colspan="2">Apply EDA and the Data Science process in a case study.</td><td>C3 (Apply)</td></tr>
<tr><td>CLO-3</td><td colspan="2">Comprehend the fundamental constructs of Python programming language.</td><td>C2 (Understand)</td></tr>
<tr><td>CLO-4</td><td colspan="2">Apply basic machine learning algorithms to solve real world problems of moderate complexity.</td><td>C3 (Apply)</td></tr>
<tr><td>Aims and Objectives</td><td colspan="3">1. Data Science is the study of the generalizable extraction of knowledge from data.
2. Being a data scientist requires an integrated skill set spanning mathematics, statistics, machine learning, databases and other branches of computer science along with a good understanding of the craft of problem formulation to engineer effective solutions.</td></tr>
</table>

	3. The aim of this course is to: Introduce students to this rapidly growing field and equip them with some of its basic principles and tools as well as its general mindset. 4. Explain the significance of exploratory data analysis in data science. Identify common approaches used for Feature Generation as well as Feature Selection, and finally discuss the Ethical and Privacy issues. 5. Programming language Python has been proposed for the practical work of this course.
Learning Outcomes	• Describe what Data Science is and the skill sets needed to be a data scientist. • Apply EDA and the Data Science process in a case study. • Comprehend the fundamental constructs of Python programming language. • Apply basic machine learning algorithms to solve real world problems of moderate complexity.
Syllabus	Introduction: What is Data Science? Big Data and Data Science hype, Datafication, Current landscape of perspectives, Skill sets needed; Statistical Inference: Populations and samples, Statistical modeling, probability distributions, fitting a model, Intro to Python; Exploratory Data Analysis and the Data Science Process; Basic Machine Learning Algorithms: Linear Regression, k-Nearest Neighbors (k-NN), k-means, Naive Bayes; Feature Generation and Feature Selection; Dimensionality Reduction: Singular Value Decomposition, Principal Component Analysis; Mining Social-Network Graphs: Social networks as graphs, Clustering of graphs, Direct discovery of communities in graphs, Partitioning of graphs, Neighborhood properties in graphs; Data Visualization: Basic principles,

	ideas and tools for data visualization; Data Science and Ethical Issues: Discussions on privacy, security, ethics, Next-generation data scientists.
Contents	1. Introduction: What is Data Science? Big Data and Data Science hype, Datafication, Current landscape of perspectives, Skill sets needed; 2. Statistical Inference: Populations and samples, Statistical modeling, probability distributions, fitting a model; 3. Intro to Python; 4. Exploratory Data Analysis and the Data Science Process; 5. Basic Machine Learning Algorithms: Linear Regression, k-Nearest Neighbors (k-NN), k-means, Naive Bayes; 6. Feature Generation and Feature Selection; 7. Dimensionality Reduction: Singular Value Decomposition, Principal Component Analysis; 8. Mining Social-Network Graphs: Social networks as graphs, Clustering of graphs, Direct discovery of communities in graphs, Partitioning of graphs, Neighborhood properties in graphs; 9. Data Visualization: Basic principles, ideas and tools for data visualization; 10. Data Science and Ethical Issues: Discussions on privacy, security, ethics, Next-generation data scientists.
Teaching-learning Strategies	• Interactive class session • Hands on practices in class • Brainstorming and Group discussion sessions • Coding in LABS
Assignments	• Paper based written assignments 4 • Coding assignments in Python 6
	Sr. # | Elements | Weightage | Details

Assessment and Examinations	1	Formative Assessment	25%	It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.
	2	Midterm Assessment	35%	It takes place at the mid-point of the semester.
	3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.
Textbooks	• An Introduction to Data Science, Jeffrey S. Saltz, Jeffrey M. Stanton, SAGE Publications, 2017. • Python for everybody: Exploring data using Python 3, Severance, C.R., CreateSpace Independent Pub Platform. 2016.			
Reference Material/Suggested Readings	• Foundations of data science, Blum, A., Hopcroft, J., & Kannan, R., Vorabversion eines Lehrbuchs, 2016. • Doing Data Science, Straight Talk from the Frontline, Cathy O'Neil and Rachel Schutt, O'Reilly. 2014. • Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data, EMC Education Services, John Wiley & Sons, 2015.			

Notes	• Academic integrity is expected of all students. Plagiarism or cheating in any assessment will result in at least an F grade in the course, and possibly more severe penalties. • You bear all the responsibility for protecting your assignments from plagiarism. If anyone else submits your assignment or uses your code in his/her assignment, you will be considered equally responsible. • The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester. • Introductory knowledge of using the computers is assumed for this course. All code written in quizzes, assignments, homework's, and exams must be in JavaScript. Code must be intelligently documented (commented). Undocumented code may not be given any credit. • The IDE use is not allowed, Notepad++ has to be used for coding. • There is no makeup for a missed sessional grading instruments like quizzes, assignments, and homework's.

Program	BS Data Science		
Course Code	DD-322		
Course Title	Data Mining and Machine Learning		
Credit Hours	**Theory**		**Lab**
	2		1
Lecture Duration	60 minutes (1 Hours), 2 lectures per week, and 3 hours Lab work		
Semester	6		
Pre-requisites	**Courses**		**Knowledge**
	DS-201 Advance Statistics, DS-302 Introduction to Data Science		Nil
Follow Up Courses	Nil		
Course Learning Outcomes (CLOs)			
CLO No	**Course Learning Outcome**		**Bloom Taxonomy**
CLO-1	Apply preprocessing techniques on any given raw data.		C3 (Apply)
CLO-2	Select and apply proper data mining algorithm to discover interesting patterns.		C3 (Apply)
CLO-3	Analyze and extract patterns to solve problems and point out how to deploy solution		C4 (Analyze)
CLO-4	Evaluate systematically supervised, semi supervised and unsupervised models and algorithms with respect to their accuracy		C4 (Analyze)
Aims and Objectives	1. The purpose of this course is to expand on the student's understanding and awareness of the concepts of data mining basics, techniques, and application. 2. The course aims to introduce the concepts of data pre-processing and Summary Statistics.		

	3. The objective of this course is to teach students the concepts of Frequent Item Set Generation, Associations and Correlations measures, Classification, Prediction, and Clustering algorithms, to apply to practical problem solving.
Learning Outcomes	• Apply preprocessing techniques on any given raw data. • Select and apply proper data mining algorithm to discover interesting patterns • Analyze and extract patterns to solve problems and point out how to deploy solution • Evaluate systematically supervised, semi supervised and unsupervised models and algorithms with respect to their accuracy
Syllabus	Introduction to data mining and basic concepts, Pre-Processing Techniques & Summary Statistics, Association Rule mining using Apriori Algorithm and Frequent Pattern Trees, Introduction to Classification Types, Supervised Classification (Decision trees, Naïve Bayes Classification, K-Nearest Neighbors, Support Vector Machines etc.), Unsupervised Classification (K Means, K Median, Hierarchical and Divisive Clustering, Kohonen Self Organizing maps), outlier & anomaly detection, Web and Social Network Mining, Data Mining Trends and Research Frontiers. Implementing concepts using Python
Contents	Unit 1: Introduction 1.1 What is Data Mining 1.2 What kinds of data can be mined 1.3 What kind of patterns can be mined 1.4 Summary statistics Unit 2: Data Pre-processing 2.1 Data Cleaning

	2.2 Data Integration 2.3 Data Reduction 2.4 Data Transformation 2.5 Data Discretization Unit 3: Mining Frequent Patterns and Associations Rules 3.1 Apriori algorithm 3.2 Generating association rules from frequent itemsets 3.3 Frequent pattern growth for finding frequent itemsets Unit 4: Classification 4.1 Basics and types 4.2 Supervised classification 4.3 Supervised classification models: Decision Trees, Naïve Bayes, Model evaluation and selection 4.4 Techniques to improve model performance Unit 5: Clustering 5.1 What is Clustering 5.2 Cluster Analysis 5.3 Partitioning methods (k-means, k-mediods) 5.4 Hierarchical methods Unit 6: Outlier and anomaly detection 6.1 What are outliers and how they affect data 6.2 Outlier analysis 6.3 Outlier detection methods Unit 7: Recent Trends 7.1 Web and Social Network Mining

	7.2 Research frontiers
Teaching-learning Strategies	• Multimedia presentations involving interactive sessions with students • Hands on exercises for concept reinforcement • Coding in laboratory
Assignments	There would be 4 programming assignments (equally divided between pre and post midterm)
Assessment and Examinations	(see table below)

Sr. #	Elements	Weightage	Details
1	Formative Assessment	25%	It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.
2	Midterm Assessment	35%	It takes place at the mid-point of the semester.
3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.

Textbooks	Han, J., Kamber, M., Pei, J. (2011). Data Mining: Concepts and Techniques. (3rd Edition). Morgan Kaufmann Publishers. ISBN 978-0-12-381479-1

Reference Material/Suggested Readings	• Tan, P., Steinbach, M., Karpatne, A., Kumar V. (2019). Introduction to Data Mining (2nd Edition). New York: Pearson Education, Inc. ISBN-13: 9780133128901 • Aggarwal, C. C. (2015). Data Mining: The Textbook. Springer International Publishing. ISBN-13: 978-3319141411 • Hand, D., Mannila, H., Smyth, P. (2001). Principles of Data Mining. MIT Press. ISBN-13: 978-0262082907 • Pattern Recognition & Machine Learning, 1st Edition, Chris Bishop • Machine Learning: A Probabilistic Perspective, 1st Edition, Kevin R Murphy • Applied Machine Learning, online Edition, David Forsyth
Notes	• Academic integrity is expected of all students. Plagiarism or cheating in any assessment will result in at least an F grade in the course, and possibly more severe penalties • There is no makeup for a missed sessional grading instruments like quizzes, assignments, and homework • The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester

Detailed Lecture wise plan

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
1	1	Introduction	Ch#1	
	2	Getting to know the data	Ch#2	
2	3	Statistical descriptors of data		Reading-1
	4	Data visualization; similarity measures		Quiz-1
3	5	Data Pre-processing: cleaning, integration	Ch#3	
	6	Data reduction		
4	7	Data Transformation and discretization		Assignment-1
	8	Mining frequent patterns: basics, market basket analysis	Ch#6	
5	9	Frequent itemsets, closed itemsets, association rules; Mining frequent itemsets		Reading-2
	10	Apriori algorithm, improving the efficiency of apriori algorithm		
6	11	Pattern growth approach to mining frequent itemsets; FP-tree		
	12	FP-growth algorithm		
7	13	Mining frequent itemsets for vertical data format		Assignment-2
	14	Mining closed and max pattens		
8	15	Pattern evaluation methods		Quiz-2
	16	Midterm review		
Midterm Exam				
9	17	Classification: introduction, types	Ch#8	Reading-3
	18	Supervised classification: Decision tree induction algorithm		
10	19	Attribute selection metrics		
	20	Tree pruning, scalability issues		Assignment-3
11	21	Bayes classification: Naïve Bayes		
	22	Model evaluation and selection		
12	23	Techniques to improve model performance		Reading-4
	24	Clustering: basics, cluster analysis	Ch#10	Quiz-3
13	25	Partitioning methods: K-means algorithm		
	26	K-medoids algorithm		Reading-5
14	27	Hierarchical clustering methods		
	28	Outiers and its types	Ch#12	Assignment-4
15	29	Outlier analysis		Quiz-4
	30	Outlier detection methods		
16	31	Recent trends	Ch#13	Reading-6

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
	32	Final term review		
		Final Exam		

<table>
<tr><td>Program</td><td colspan="3">BS data science</td></tr>
<tr><td>Course Code</td><td colspan="3">DD-321</td></tr>
<tr><td>Course Title</td><td colspan="3">Data Warehousing and Business Intelligence</td></tr>
<tr><td rowspan="2">Credit Hours</td><td colspan="2">Theory</td><td>Lab</td></tr>
<tr><td colspan="2">2</td><td>1</td></tr>
<tr><td>Lecture Duration</td><td colspan="3">60 minutes (1 Hours), 2 lectures per week</td></tr>
<tr><td>Semester</td><td colspan="3">5</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td colspan="2">Courses</td><td>Knowledge</td></tr>
<tr><td colspan="2">Introduction to Data Science</td><td>Nil</td></tr>
<tr><td>Follow Up Courses</td><td colspan="3">Big Data Analytics</td></tr>
<tr><td colspan="4">Course Learning Outcomes (CLOs)</td></tr>
<tr><td>CLO No</td><td colspan="2">Course Learning Outcome</td><td>Bloom Taxonomy</td></tr>
<tr><td>CLO-1</td><td colspan="2">Demonstrate an appreciation of the role that Data Warehouses and Business Intelligence play in enhancing the decision-making process.</td><td>C2 (Understand)</td></tr>
<tr><td>CLO-2</td><td colspan="2">Demonstrate an understanding of the fundamental concepts of the Star and the Snowflake Schema; learn how to design the schema of a DW based on these two models.</td><td>C2 (Understand)</td></tr>
<tr><td>CLO-3</td><td colspan="2">Understand the architecture of DW Systems and be able to specify the advantages and potential problem areas</td><td>C3 (Apply)</td></tr>
<tr><td>CLO-4</td><td colspan="2">Use Analytic SQL to aggregate, analyze and report, and model data.</td><td>C3 (Apply)</td></tr>
<tr><td>Aims and Objectives</td><td colspan="3">Upon completion of this course, the student will be able to:
1. Grasp modeling of data warehouse
2. Design Data Warehouse
3. Implement Data Warehouse and BI
•</td></tr>
<tr><td>Learning Outcomes</td><td colspan="3">At the end of the course, you should be able to:</td></tr>
</table>

	• Demonstrate an appreciation of the role that Data Warehouses and Business Intelligence play in enhancing the decision-making process • Demonstrate an understanding of the fundamental concepts of the Star and the Snowflake Schema; learn how to design the schema of a DW based on these two models. • Understand the architecture of DW Systems and be able to specify the advantages and potential problem areas • Use Analytic SQL to aggregate, analyze and report, and model data.
Contents	1. Introduction to Data Warehouse and Business Intelligence 1.1. Necessities of BI 1.2. Essentials of Business Intelligence 1.3. DW Life Cycle 1.4. Basic Architecture 2. DW Architecture in SQL Server 2.1. Logical Model 2.2. Indexes 2.3. Physical Model 2.4. Optimizations 3. OLAP Operations 3.1. Queries 3.2. Query Optimization 4. Building the DW 4.1. Data visualization 4.2. Reporting based on Data Warehouse 4.3. Reporting using SSAS 4.4. Reporting using Tableau 5. Data visualization 5.1. Reporting with visualization 5.2. Reporting based on Cube 5.3. Reports management 6. Dashboard management

<table>
<tr><td></td><td>6.1. PowerBI Dashboard
6.2. Dashboard Enrichment
7. Business Intelligence Tools.</td></tr>
<tr><td>Teaching-learning Strategies</td><td>• Interactive class session
• Hands on practices in class
• Brainstorming and Group discussion sessions</td></tr>
<tr><td>Assignments</td><td>• Paper based written assignments 3
• Project 2
• Quiz 4</td></tr>
<tr><td>Assessment and Examinations</td><td>
<table>
<tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper.</td></tr>
</table>
</td></tr>
<tr><td>Textbooks & Reference material</td><td>• W. H. Inmon, “Building the Data Warehouse”, Wiley-India Edition.
• 2. Ralph Kimball, “The Data Warehouse Toolkit – Practical Techniques for Building Dimensional Data Warehouse,” John Wiley & Sons, Inc.
• 3. Matteo Golfarelli, Stefano Rizzi, “Data Warehouse Design - Modern Principles and Methodologies”, McGraw Hill Publisher</td></tr>
</table>

Notes	• The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester.

<table>
<tr><td>Program</td><td colspan="3">BS DS</td></tr>
<tr><td>Course Code</td><td colspan="3">DD-222</td></tr>
<tr><td>Course Title</td><td colspan="3">Data Visualization & Graphics</td></tr>
<tr><td rowspan="2">Credit Hours</td><td colspan="2">Theory</td><td>Lab</td></tr>
<tr><td colspan="2">2</td><td>1</td></tr>
<tr><td>Lecture Duration</td><td colspan="3">60 minutes (1 Hours), 2 lectures per week, 3 hours lab session per week</td></tr>
<tr><td>Semester</td><td colspan="3">5</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td colspan="2">Courses</td><td>Knowledge</td></tr>
<tr><td colspan="2"></td><td>Nil</td></tr>
<tr><td>Follow Up Courses</td><td colspan="3">Nill</td></tr>
<tr><td colspan="4">Course Learning Outcomes (CLOs)</td></tr>
<tr><td>CLO No</td><td colspan="2">Course Learning Outcome</td><td>Bloom Taxonomy</td></tr>
<tr><td>CLO-1</td><td colspan="2">Provides knowledge about importance, necessity, and justification of performing exploratory data analysis and visualization</td><td>C2 (Understand)</td></tr>
<tr><td>CLO-2</td><td colspan="2">Introduce various type of charts along with their alternatives solution to show same data from versatile aspects.</td><td>C2 (Understand)</td></tr>
<tr><td>CLO-3</td><td colspan="2">Improving the competency of the students to analyze different problems and select the most appropriate solution.</td><td>C3 (Apply)</td></tr>
<tr><td>CLO-4</td><td colspan="2">Use of R, various recent tools, and technologies to develop hands-on skills for exploratory data analysis and visualization.</td><td>C3 (Apply)</td></tr>
<tr><td>Aims and Objectives</td><td colspan="3">1. Provides knowledge about importance, necessity, and justification of performing exploratory data analysis and visualization
2. Introduce various type of charts along with their alternatives solution to show same data from versatile aspects</td></tr>
</table>

	3. Improving the competency of the students to analyze different problems and select the most appropriate solution 4. Use of Python, various recent tools, and technologies to develop hands-on skills for exploratory data analysis and visualization
Learning Outcomes	• Provides knowledge about importance, necessity, and justification of performing exploratory data analysis and visualization • Introduce various type of charts along with their alternatives solution to show same data from versatile aspects • Improving the competency of the students to analyze different problems and select the most appropriate solution • Use of Python, various recent tools, and technologies to develop hands-on skills for exploratory data analysis and visualization
Syllabus	Introduction of Exploratory Data Analysis and Visualization, Building Blocks and Basic Operations; Types of Exploratory Graphs, single and multi-dimensional summaries, five number summary, box plots, histogram, bar plot and others; Distributions, their representation using histograms, outliers, variance; Probability Mass Functions and their visualization; Cumulative distribution functions, percentile-based statistics, random numbers; Modelling distributions, exponential, normal, lognormal, pareto; Probability density functions, kernel density estimation; Relationship between variables, scatter plots, correlation, covariance; Estimation and Hypothesis Testing; Clustering using K-means and Hierarchical; Time series and survival analysis; Implementing concepts with Python
Contents	1. Introduction of Exploratory Data Analysis and Visualization, 2. Building Blocks and Basic Operations; 3. Types of Exploratory Graphs 4. Single and multi-dimensional summaries, 5. Five number summary

	5.1. box plots, 5.2. histogram, 5.3. bar plot and others; 6. Distributions, their representation using histograms, 7. Outliers, variance; 8. Probability Mass Functions and their visualization; 9. Cumulative distribution functions, 10. Percentile-based statistics, 11. Random numbers; 12. Modelling distributions, 12.1. Exponential, 12.2. Normal, 12.3. Lognormal, 12.4. Pareto. 13. Probability density functions, 14. Kernel density estimation; 15. Relationship between variables, 15.1. Scatter plots, 15.2. Correlation, 15.3. Covariance; 15.4. Estimation and Hypothesis Testing; 15.5. Clustering using K-means and Hierarchical; 15.6. Time series and survival analysis 15.7. Implementing concepts with Python
Teaching-learning Strategies	• Interactive class session • Hands on practices in class • Brainstorming and Group discussion sessions
Assignments	• Programming Assignments 8

<table>
<tr><td rowspan="4">Assessment and Examinations</td><td>Sr. #</td><td>Elements</td><td>Weightage</td><td>Details</td></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.</td></tr>
<tr><td>Textbooks</td><td colspan="4">Interactive Data Visualization with Python: Present your data as an effective and compelling story, 2nd Edition, Abha Belorkar, Sharath Chandra Guntuku, Shubhangi Hora, Anshu Kumar
Data Visualization in Python, Daniel Nielson</td></tr>
<tr><td>Reference Material/Suggested Readings</td><td colspan="4">Handout provided by the teacher.
PowerPoint Presentations
Various books Chapters / Notes
Internet resources</td></tr>
</table>

Notes	• Academic integrity is expected of all students. Plagiarism or cheating in any assessment will result in at least an F grade in the course, and possibly more severe penalties. • You bear all the responsibility for protecting your assignments from plagiarism. If anyone else submits your assignment or uses your code in his/her assignment, you will be considered equally responsible. • The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester. • There is no makeup for a missed sessional grading instruments like quizzes, assignments, and homework's.

<table>
<tr><td>Program</td><td colspan="3">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="3">ED-421</td></tr>
<tr><td>Course Title</td><td colspan="3">Big Data Analytics</td></tr>
<tr><td rowspan="2">Credit Hours</td><td colspan="2">Theory</td><td>Lab</td></tr>
<tr><td colspan="2">2</td><td>1</td></tr>
<tr><td>Lecture Duration</td><td colspan="3">60 minutes (1 Hours) 2 lectures per week, 3 hours Lab per week</td></tr>
<tr><td>Semester</td><td colspan="3">7</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td colspan="2">Courses</td><td>Knowledge</td></tr>
<tr><td colspan="2"></td><td>Basic statistics and programming knowledge is required</td></tr>
<tr><td>Follow Up Courses</td><td colspan="3"></td></tr>
<tr><td colspan="4">Course Learning Outcomes (CLOs)</td></tr>
<tr><td>CLO No</td><td colspan="2">Course Learning Outcome</td><td>Bloom Taxonomy</td></tr>
<tr><td>CLO-1</td><td colspan="2">Understand the fundamental concepts of Big Data and its programming paradigm.</td><td>C2 (Understand)</td></tr>
<tr><td>CLO-2</td><td colspan="2">Hadoop/MapReduce Programming, Framework, and Ecosystem</td><td>C3 (Apply)</td></tr>
<tr><td>CLO-3</td><td colspan="2">Apache Spark Programming</td><td>C3 (Apply)</td></tr>
<tr><td>Objectives</td><td colspan="3">1. The main goal of this course is to help students learn, understand, and practice applications for big data analytics and machine learning approach.
2. Students will get hands-on experience in applying big data technologies and machine learning techniques to industry applications.</td></tr>
</table>

Learning Outcomes	• *Ability to identify the characteristics of datasets* • *Ability to select and implement machine learning techniques for different type of applications* • *Solve problems associated with batch learning and online learning, and the big data characteristics such as high dimensionality, dynamically growing data and in particular scalability issues* • *Ability to integrate machine learning libraries with modern technologies like hadoop and mapreduce.*
Contents	*1. Introduction to Big Data Analytics* *1.1 Concept, History, and Trends* *2. Big Data Platforms* *2.1 Hadoop, Spark, and NoSQL Stores* *3. Data Store & Processing using Hadoop* *3.1 Introduction to MapReduce* *3.2 Programming Hadoop* *4. Big Data Storage and Analytics* *4.1 HDFS* *4.2 NoSQL databases* *5. Big Data Analytics using ML Algorithms* *5.1 Introduction to Predictive Analytics* *5.2 Machine Learning Algorithm Types* *5.3 Big Data and ML* *6. Recommendation Systems* *6.1 Introduction* *6.2 Type of Recommendation System*

<table>
<tr><td></td><td>

6.3 Recommendation Systems using Big

7. Supervised and Unsupervised Learning

7.1 Introduction to Supervise Learning

7.2 Supervised Learning Algorithms

7.3 Introduction to Unsupervised Learning

7.4 Unsupervised Learning Algorithms

7.5 Cases Studies for Supervised and Unsupervised Learning

8. Linked Big Data: Graph Computing and Graph Analytics

9. Big Data Visualization

10. Dimensionality Reduction in Big Data

11. Big Data Applications in Healthcare, IoT, and Smart Cities

11.1 Case Studies for Big Data in different Industries

12. Research topics in Big Data Analytics

</td></tr>
<tr><td>

Teaching-learning Strategies

</td><td>

- *Interactive class session*
- *Hands on assignments and tutorials*
- *Group project*

</td></tr>
<tr><td>

Assignments

</td><td>

- *Practical Assignments 5*

</td></tr>
<tr><td>

Assessment and Examinations

</td><td>
<table>
<tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
</table>
</td></tr>
</table>

	2	*Midterm Assessment*	*35%*	*It takes place at the mid-point of the semester.*
	3	*Final Assessment*	*40%*	*It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.*
Reference Material/Suggested Readings	• *Leskovec, J., Rajaraman, A., & Ullman, J. D. (2020). Mining of massive data sets. Cambridge university press.* • *White, T. (2012). Hadoop: The definitive guide. O'Reilly Media, Inc.* • *Lin, J., & Dyer, C. (2010). Data-intensive text processing with MapReduce. Synthesis Lectures on Human Language Technologies, 3(1), 1-177.*			
Notes	• *Academic integrity is expected of all students. Plagiarism or cheating in any assessment will result in at least an F grade in the course, and possibly more severe penalties.* • *You bear all the responsibility for protecting your assignments from plagiarism. If anyone else submits your assignment or uses your code in his/her assignment, you will be considered equally responsible.* • *The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester.*			

	• *There is no makeup for a missed sessional grading instruments like quizzes, assignments, and homework's.*

Detailed Lecture wise plan

Week	***Lecture***	***Topic***	***SourceBook (Ch#)***	***Recommendation for Learning Activities***
1	*1*	*Introduction to Big Data Analytics*		
	2	*Concept, History, and Trends*		*Assignment*
2	*3*	*Big Data Platforms*		
	4	*Hadoop, Spark, and NoSQL Stores*		*Quiz*
3	*5*	*Data Store & Processing using Hadoop*		
	6	*Introduction to MapReduce*		
4	*7*	*Programming Hadoop*		*Assignment*
	8	*Big Data Storage and Analytics*		
5	*9*	*HDFS*		*Quiz*
	10	*NoSQL databases*		
6	*11*	*Big Data Analytics using ML Algorithms*		*Assignment*
	12	*Introduction to Predictive Analytics*		
7	*13*	*Machine Learning Algorithm Types*		
	14	*Big Data and ML*		*Assignment*
8	*15*	*Introduction to Recommendation Systems*		
	16	*Type of Recommendation System*		
Midterm Exams				
9	*17*	*Recommendation Systems using Big Data*		*Quiz*

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
	18	Importance of Supervised and Unsupervised Learning		
10	19	Introduction to Supervise Learning		
	20	Supervised Learning Algorithms		
11	21	Introduction to Unsupervised Learning		
	22	Unsupervised Learning Algorithms		Assignment
12	23	Cases Studies for Supervised and Unsupervised Learning		
	24	Linked Big Data: Graph Computing and Graph Analytics		Quiz
13	25	Big Data Visualization		
	26	Dimensionality Reduction in Big Data		
14	27	Big Data Applications in Healthcare, IoT, and Smart Cities		
	28	Big Data Applications in Healthcare, IoT, and Smart Cities (Cont.)		Quiz
15	29	Case Studies for Big Data in different Industries		
	30	Research topics in Big Data Analytics		

Week	***Lecture***	***Topic***	***SourceBook (Ch#)***	***Recommendation for Learning Activities***
16	*31*	*Research topics in Big Data Analytics (Cont.)*		*Research Papers Reading*
	32	*Research topics in Big Data Analytics (Cont.)*		*Research Papers Reading*
Final Exam				

<table>
<tr><td>Program</td><td colspan="2">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="2">MS-251</td></tr>
<tr><td>Course Title</td><td colspan="2">Probability and Statistics</td></tr>
<tr><td rowspan="2">Credit Hours</td><td>Theory</td><td>Lab</td></tr>
<tr><td>3 Cr Hrs</td><td>0</td></tr>
<tr><td>Lecture Duration</td><td colspan="2">90 minutes (1.5 Hours), 2 lectures per week</td></tr>
<tr><td>Semester</td><td colspan="2">2</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td>Courses</td><td>Knowledge</td></tr>
<tr><td></td><td></td></tr>
<tr><td>Follow Up Courses</td><td colspan="2">Advanced Statistics</td></tr>
</table>

<table>
<tr><td colspan="3">Course Learning Outcomes (CLOs)</td></tr>
<tr><td>CLO No</td><td>Course Learning Outcome</td><td>Bloom Taxonomy</td></tr>
<tr><td>CLO-1</td><td>Acquire knowledge related to the concepts, tools and techniques for the design of digital electronic circuits</td><td>C2 (Understand)</td></tr>
</table>

<table>
<tr><td>Aims and Objectives</td><td>1. Students will get a strong foundation of probability and statistics.
2. Students will get in depth knowledge of how to do statistical analysis and apply them in real life scenario.
3. Students should be able to apply the learned concepts in MATLAB.
4. After the completion of this course students should be able to conduct research based surveys. It is expected from a student to submit a paper in a conference or journal based on their project.</td></tr>
<tr><td>Learning Outcomes</td><td>•</td></tr>
<tr><td>Syllabus</td><td>Topics: Introduction to Statistics, Pure Probability, Random Variables (Discrete and Continuous, Joint Random Variables), Probability Distribution (Discrete and Continuous, Binomial, Hypergeometric, Poisson, Normal distributions.),</td></tr>
</table>

	Sampling, Sampling Distributions (for mean large and small sample, difference between means for large and small samples, proportions, t distribution), Estimation point and confidence interval estimation (mean for large and small sample, difference between mean for large and small samples, proportions), Hypothesis Testing (for mean, difference between mean, proportion, independence of variables). Three sigma rule, law of large numbers, Simulation and application of all learned the concepts in MATLAB.
Contents	***BOOK(A):*** ***Chapter 1: Introduction to Statistics and Data Analysis*** *1.3 Measures of location* *1.4 Measures of variability* ***Chapter 3: Concept of a Random Variable and Discrete Probability Distributions*** *3.1 Concepts of random variable* *3.2 Discrete probability distributions* ***Chapter 5: Some Discrete Probability Distributions*** *5.3 Hypergeometric distribution* *5.4 Negative binomial and geometric distribution* *5.5 Poisson distribution and the Poisson process* ***Chapter 6: Some Continuous Probability Distributions*** *6.1 Continuous uniform distribution* *6.2 Normal distribution* *6.3 Areas under the normal curve* *6.4 Application of the normal distribution* *6.5 Normal approximation to the normal* ***Chapter 8: Fundamental Sampling Distribution and Data Description*** *8.1 Hypergeometric distribution* *8.2 Negative binomial and geometric distribution* *8.3 Poisson distribution and the Poisson process*

	8.4 Sampling distribution of mean and central limit theorem ***Chapter 9: One- and Two- Sample Estimation Problems*** *9.1 Introduction* *9.2 Statistical inference* *9.3 Classical methods of estimation* ***Chapter 10: One- and Two- Sample Tests of Hypothesis*** *10.1 Statistical hypothesis* *10.2 Testing a statistical hypothesis* *10.4 Single sample* *10.5 Two samples* *10.8 One sample* *10.9 Two samples* ***BOOK(B):*** ***Chapter 1: Basic Concepts:*** • *Course introduction* • *Probability* • *probability experiment* • *outcome, trial* • *sample space* • *random* • *equally likely* • *event* • *simple event* • *compound event* • *classical probability* • *probability rules* ***Chapter 2: Sample Spaces:*** • *Impossible event*

	• *Unlikely event* • *50-50 event* • *Likely event* • *Certain event* • *Complement of an event* • *Frequency distributions* • *Empirical probability* • *Sample, population* • *Law of large numbers* • *Subjective probability* • *Tree diagrams* • *Tables* ***Chapter 3: The Addition Rules:*** • *Addition rules* • *Mutually exclusive event* • *Not mutually exclusive event* • *Addition Rule I* • *Addition Rule II* ***Chapter 4: The Multiplication:*** • *The multiplication rules* • *Independent events* • *Dependent events* • *Multiplication Rule I* • *Multiplication Rule II* • *Conditional probability* • *Alternative approach of calculating conditional probability* ***Chapter 7: The Binomial Distribution***

<table>
<tr><td></td><td>

- *Binomial distribution*
- *Binomial experiment*
- *Mean*
- *Standard deviation*
- *Variance of binomial distribution*
- *Calculating probabilities using formula*
- *Tables*

Chapter 8: Other Probability Distribution

- *Hypergeometric distribution*
- *Hypergeometric experiment*
- *Mean*
- *Standard deviation*
- *Variance of hypergeometric distribution*
- *Calculating probabilities using formula*
- *Table*

MATLAB. Relationship to the binomial distribution.

Chapter 1:
</td></tr>
<tr><td>Teaching-learning Strategies</td><td>

-
</td></tr>
<tr><td>Assignments</td><td>Types and Number with calendar</td></tr>
<tr><td>Assessment and Examinations</td><td>
<table>
<tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework,</td></tr>
</table>
</td></tr>
</table>

				attitude and behavior, hands-on-activities, short tests, quizzes etc.
	2	*Midterm Assessment*	*35%*	*It takes place at the mid-point of the semester.*
	3	*Final Assessment*	*40%*	*It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.*
Textbooks	*A. R.E. Walpole, R.H. Myers and S.L Myers, "Probability and Statistics for Engineers and Scientists'', Prentice Hall, 9th Edition* *B. Allan G. Bluman, "Probability Demystified'', McGraw-Hill Professional, 1st Edition (January 27, 2005)* *C. Larry J. Stephens, "Advanced Statistics Demystified'', McGraw-Hill Professional, 1st Edition (June 1, 2004)*			
Reference Material/Suggested Readings	*R1. Valerie M. Sue and Lois A. Ritter, "Conducting Online Surveys'', SAGE Publications, Inc, 2nd Edition (November 23, 2011)* *R2. Vivek Bhaskaran and Jennifer LeClaire, "Online Surveys For Dummies'', For Dummies, 1st Edition (June 15, 2010)* *R3. David McMahon, "MATLAB Demystified'', McGraw-Hill Professional, 1st Edition (April 6, 2007)* *R4. Timothy A. Davis Kermit Sigmon, "MATLAB® Primer'', CRC Press, 8th Edition (August 18, 2010)*			
Notes	• *MATLAB* • *Slides for each topic*			

Detailed Lecture wise plan

Week	***Lecture***	***Topic***	***Sourcebook (Ch#)***	***Recommendation for Learning Activities***
1	*1*	*Course introduction, **probability**, probability experiment, outcome, trial, sample space, random, equally likely, event, simple event, compound event, classical probability, and probability rules*	*B. Chapter 1: Basic concepts*	
	2	*Impossible event, unlikely event, 50-50 event, likely event, certain event, complement of an event, frequency distributions, empirical probability, sample, population, law of large numbers, subjective probability, tree diagrams, and tables*	*B. Chapter 2: Sample Spaces*	
2	*3*	*Addition rules, mutually exclusive event, not mutually exclusive event, addition Rule I, and addition Rule II* *MATLAB: Introduction to MATLAB, command window and basic arithmetic*	*B. Chapter 3: The Addition Rules* *R4. Chapter 1: Getting Started* *R4. Chapter 2: The MATLAB Desktop* *R4. Chapter 3: Matrices and Matrix Operations*	
	4	*The multiplication rules, independent events, dependent events, Multiplication Rule I, and Multiplication Rule II* *MATLAB: Referencing individual entries, relational operators, logical operators, simulation of coin tosses, simulation of the sum of two fair dice*	*B. Chapter 4: The Multiplication Rules* *R4. Chapter 3: Matrices and Matrix Operations*	
3	*5*	*Conditional probability, alternative approach of calculating conditional probability.*	*B. Chapter 4: The Multiplication Rules*	***Assign-1***

Week	Lecture	Topic	Sourcebook (Ch#)	Recommendation for Learning Activities
	6	Concepts of random variable, probability distribution, discrete probability distributions, properties of discrete probability distributions, types of discrete probability distributions.	A. Chapter 3: Concept of a Random Variable and Discrete Probability Distributions B. Chapter 7: The Binomial Distribution B. Chapter 1, 2, 3, and 4	**Quiz#1**
4	7	Binomial distribution, binomial experiment, mean, standard deviation, and variance of binomial distribution	B. Chapter 7: The Binomial Distribution	
	8	Binomial distribution: calculating probabilities using formula, tables, and MATLAB. binomial approximation to Gaussian distribution. MATLAB: **binopdf**	B. Chapter 7: The Binomial Distribution A. Chapter 5.2	
5	9	Hypergeometric distribution, hypergeometric experiment, mean, standard deviation, and variance of hypergeometric distribution.	B. Chapter 8: Other Probability Distribution A. Chapter 5.3	
	10	Hypergeometric distribution: calculating probabilities using formula, table, and MATLAB. Relationship to the binomial distribution.	B. Chapter 8: Other Probability Distributions A. Chapter 5.3	**Quiz#2**
6	11	Binomial approximation to hypergeometric distribution. MATLAB: hygepdf(x, N, k, n). The 68-95-99.7 rule - or three-sigma rule, or empirical rule.		
	12	A. Chapter 5.3	A. Chapter 5.5	
7	13	Mean, standard deviation, and variance of the Poisson distribution	A. Chapter 5.5	

Week	***Lecture***	***Topic***	***Sourcebook (Ch#)***	***Recommendation for Learning Activities***
	14	*Nature of the Poisson probability function, binomial distribution approximation to Poisson distribution.*	*A. Chapter 5.5 R1-Reading Material*	***Assign-2***
8	*15*	*Tutorial on conducting research based surveys.*	*R1*	***Quiz#3***
	16	*Some continuous probability distributions: continuous uniform distribution, pdf, mean, and variance* ***MATLAB: unifpdf***	*A. Chapter 6.1*	
Mid Term Examination				
9	*17*	*Some continuous probability distributions: Normal distribution, pdf, properties of the normal distribution, mean, and variance*	*A. Chapter 6.2*	*Project proposal submission deadline*
	18	*Normal curves with different combination of values of mean and standard deviation* *Areas under the normal curve, standard normal distribution.*	*A. Chapter 6.3*	
10	*19*	*Using the normal curve in reverse*	*A. Chapter 6.3*	
	20	*Applications of the normal distribution*	*A. Chapter 6.4*	*Project questionnaire submission deadline*
11	*21*	*Normal approximation to binomial distribution* *Matlab: normpdf*	*A. Chapter 6.5*	
	22	*Measures of location: the sample mean and median*	*A-Chapter 1.3*	***Assign-3***

Week	Lecture	Topic	Sourcebook (Ch#)	Recommendation for Learning Activities
12	23	Other measures of locations: trimmed means Measures of variability: sample range and sample standard deviation, units for standard deviation and variance, which variability measure is more important?	A-Chapter 1.4	Project data collection deadline
	24	Fundamental sampling distributions and data descriptions: random sampling, populations and samples .	A-Chapter 1.4	**Quiz#4**
13	25	Some important statistics: location measures of a sample: the sample mean, median, and mode. Variability measures of a sample: the sample variance, standard deviation, And range	A-Chapter 8.1-8.2	
	26	Sampling distributions: inference about the population from sample information Sampling distribution of means and the central limit theorem	A-Chapter 8.3 A-Chapter 8.4	
14	27	One- and Two-Sample Estimation Problems**:** Statistical Inference**,** Classical Methods of Estimation**,** Unbiased Estimator, Variance of a Point Estimator, Interval Estimation	A-Chapter 9.1, 9.2, 9.3	
	28	Single sample: estimating the mean, confidence interval on μ, σ^2 known, one-sided confidence bounds, one-sided confidence bounds on μ, σ^2 known, the case of σ unknown, confidence interval on μ, σ^2 known, concept of a large-sample confidence interval, standard error of a point estimate	A-Chapter 9.4, 9.5	**Quiz#5**

Week	*Lecture*	*Topic*	*Sourcebook (Ch#)*	*Recommendation for Learning Activities*
15	*29*	*One- and two-sample tests of hypotheses: statistical hypotheses: general concepts, the role of probability in hypothesis testing, the null and alternative hypotheses, testing a statistical hypothesis, the test statistic, the probability of a type I error, the probability of a type II error, one- and two-tailed tests, how are the null and alternative hypotheses chosen?, single sample: tests concerning a single mean, tests on a single mean (variance known), tests on a single sample (variance unknown) is based on normal distribution and estimation*	*A-Chapter 10.1, 10.2*	**Assign-4**
	30	*Two samples: tests on two means: unknown but equal variances, two-sample pooled t-test* *One sample: test on a single proportion* *Two samples: tests on two proportions*	*A-Chapter 10.4, 10.5* *A-Chapter 10.8* *A-Chapter 10.9*	**Quiz#6**
16	*31* *32*	*Project submission deadline and presentation group member.*		

Program	BS data science	
Course Code	GE-162	
Course Title	Calculus and Analytical Geometry	
Credit Hours	**Theory**	**Lab**
	3	0
Lecture Duration	90 minutes (1.5 Hours), 2 lectures per week	
Semester	3	
Pre-requisites	**Courses**	**Knowledge**
	Nil	Nil
Follow Up Courses	Differential Equations	
Aims and Objectives	1. Students should be able to work with functions represented in a variety of Ways: graphical, numerical, analytical, or verbal. They should understand the Connections among these representations. 2. Students should understand the meaning of the derivative in terms of a rate of change and local linear approximation and should be able to use derivatives to solve a variety of problems. 3. Students should understand the meaning of the definite integral both as a limit of Riemann sums and as the net accumulation of a rate of change and should be able to use integrals to solve a variety of problems	

Learning Outcomes	*At the end of the course, you should be able to:* • *Use a variety of methods in solving real-life, practical, technical, and theoretical problems.* • *Select and use an appropriate problem-solving strategy.* • *Explain the limit process and that calculus centers around this concept.* • *Identify the two classical problems that were solved by the discovery of calculus, The tangent problem and the area problem.* • *Describe the two main branches of calculus, Differential calculus and Integral calculus.*
Syllabus	*Limits and Continuity; Introduction to functions, Introduction to limits, Techniques of funding limits, Indeterminate forms of limits, Continuous and discontinuous functions and their applications, Differential calculus; Concept and idea of differentiation, Geometrical and Physical meaning of derivatives, Rules of differentiation, Techniques of differentiation, Rates of change, Tangents and Normal lines, Chain rule, implicit differentiation, linear approximation, Applications of differentiation; Extreme value functions, Mean value theorems, Maxima and Minima of a function for single-variable, Concavity, Integral calculus; Concept and idea of Integration, Indefinite Integrals, Techniques of integration, Riemann sums and Definite Integrals, Applications of definite integrals, Improper integral, Applications of Integration; Area under the curve, Analytical Geometry; Straight lines in R3, Equations for planes.*
Contents	***1. Limits and continuity*** ***1.1.*** *An intuitive approach to limits* ***1.2.*** *Two sided limits and one sided limit.* ***1.3.*** *Techniques of computing limits* ***1.4.*** *Limits at infinity* ***1.5.*** *Limit discussed more rigorously* ***1.6.*** *Introduction to continuity* ***1.7.*** *Techniques of checking continuity* ***1.8.*** *Continuity* *1.8.1. Trigonometric function* ***1.8.2.*** *Inverse Trigonometric function* ***1.9.*** *Limits and continuity of trigonometric functions* ***1.10.*** *Exponential and Logarithmic functions* ***1.11.*** *Applications of continuity* ***1.12.*** *Examples of applications* ***2. The Derivative*** ***2.1.*** *Differential calculus*

	2.1.1. *Motivation of derivatives* ***2.1.2.*** *Tangent line* ***2.2.*** *Geometrical and Physical meaning of derivatives* ***2.2.1.*** *Derivative of a function* ***2.2.2.*** *Rules of differentiation* ***2.2.3.*** *Differentiation by parts* ***2.3.*** *Concept and idea of differentiation* ***2.4.*** *Differentiation by part* ***2.4.1.*** *Techniques of differentiation* ***2.4.2.*** *Derivative of trigonometric functions* ***2.5.*** *Product and quotient rule* ***2.6.*** *Inverse Trigonometric function* ***2.7.*** *Application of derivatives as rates of change,* ***2.8.*** *Derivatives of trigonometric functions* ***2.9.*** *The chain rule* ***3. Topics in Differentiation*** ***3.1.*** *Implicit differentiation* ***3.2.*** ***3.3.*** *Derivatives of logarithmic functions* ***3.4.*** *Implicit differentiation* ***3.5.*** *Derivatives of logarithmic functions.* ***3.6.*** *Derivatives of exponential and inverse trigonometric function* ***3.7.*** *Local linear approximation,* ***3.8.*** *Differentials* *3.8.1. L'hospital rules* ***3.8.2.*** *Indeterminate form* ***4. The derivative in Graphing and application*** ***4.1.*** *Analysis of functions* ***4.1.1.*** *Increasing function* ***4.1.2.*** *Decreasing Functions* ***4.2.*** *Relative extrema* ***4.3.*** *Graphing a function* ***4.3.1.*** *Concavity* ***4.3.2.*** *Rational functions* ***4.3.3.*** *Cusps* ***4.3.4.*** *Vertical tangents* ***4.4.*** *Relative extrema* ***4.4.1.*** *Absolute maxima* ***4.4.2.*** *Absolute minima* ***4.4.3.*** *Applied Maximum and minimum problem* ***4.4.4.*** *Mean value theorem* ***4.4.5.*** *Mean value theorem*

	4.4.6. *Roll's theorem* ***5. Integration*** ***5.1. What is integration*** ***5.1.1.*** *An overview of area problems* ***5.2.*** *The indefinite integrals* ***5.2.1.*** *Integration by substitution* ***5.2.2.*** *Area as a limit* ***5.3.*** *The definite integral* ***5.3.1.*** *Riemann sums and Definite Integrals,* ***5.3.2.*** *The fundamental theorem of calculus* ***5.3.3.*** *The definite integral by substitution* ***5.3.4.*** *Transcendental functions integral* ***5.3.5.*** *An overview of integration methods* ***5.3.6.*** *Integration by parts* ***5.3.7.*** *Integration Trigonometric Functions* ***5.4. Principal of integral evaluation*** ***5.4.1.*** *An overview of integration methods* ***5.4.2.*** *Integration by parts* ***5.4.3.*** *Integration Trigonometric Functions* ***5.4.4.*** *Trigonometric substitution* ***5.4.5.*** *Integration by partial fractions* ***5.4.6.*** *Area between two curves* ***6. Three dimensional spaces*** ***6.1.*** *Parametric equations of line* ***6.2.*** *Evaluation of parametric equations of lines* ***6.3.*** *Planes in 3D space* ***6.4.*** *Distance between planes* ***6.5.*** *Distance between line and plane* ***6.6.*** *Planes in 3D space* ***6.7.*** *Distance between planes* ***6.8.*** *Distance between line and plane* ***6.9.*** *Planes in 3D space* ***6.9.1.*** *Distance between planes* ***6.9.2.*** *Distance between line and plane* ***7. Partial differentiation*** ***7.1.*** *Partial derivative concepts* ***8. Multiple integrals*** ***8.1.*** *Multiple integral concepts* ***9. Mathematical modeling with differential equations*** ***9.1.*** *Modeling with differential equations* ***9.2.*** *Types of differential equations*

<table>
<tr><td>Teaching-learning Strategies</td><td colspan="4">Hands on practices in class
Brainstorming and Group discussion sessions on applications of the topics.</td></tr>
<tr><td>Assignments</td><td colspan="4">Paper based written assignments 6
Paper based Quiz 10</td></tr>
<tr><td rowspan="4">Assessment and Examinations</td><td>Sr. #</td><td>Elements</td><td>Weightage</td><td>Details</td></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper.</td></tr>
<tr><td>Textbooks</td><td colspan="4">A. Thomas, G. B., Finney, R. L., Weir, M. D., & Giordano, F. R. (2003). Thomas' calculus. Reading: Addison-Wesley.</td></tr>
<tr><td>Reference Material</td><td colspan="4">B: Anton, H., & Nicoletti, G. (1988). Calculus (Vol. 10). New York: Wiley.
C: Zill, D. G. (2016). Differential equations with boundary-value problems. Cengage Learning.
D: Online Material: www.mathworld.com</td></tr>
</table>

Notes	• *The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester.* • *Instructor can change the order of topics to provide ease to students to understand.*

<table>
<tr><td>Program</td><td colspan="2">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="2">MS-252</td></tr>
<tr><td>Course Title</td><td colspan="2">Linear Algebra</td></tr>
<tr><td rowspan="2">Credit Hours</td><td>Theory</td><td>Lab</td></tr>
<tr><td>3</td><td>0</td></tr>
<tr><td>Lecture Duration</td><td colspan="2">90 minutes (1.5 Hours), 2 lectures per week</td></tr>
<tr><td>Semester</td><td colspan="2">3</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td>Courses</td><td>Knowledge</td></tr>
<tr><td>Nil</td><td>Nil</td></tr>
<tr><td>Follow Up Courses</td><td colspan="2">Nil</td></tr>
<tr><td>Aims and Objectives</td><td colspan="2">1. Comprehend vector spaces (subspaces).
2. Understand fundamental properties of matrices including inverse matrices, eigenvalues and linear transformations.
3. Be able to solve linear systems of equations.
4. Have an insight into the applicability of linear algebra.
5. Apply linear algebra concepts to model, solve, and analyze real-world situations.
6. Students should be able to apply the learned concepts in MATLAB.</td></tr>
<tr><td>Learning Outcomes</td><td colspan="2">• After the completion of this course students should get the right background to study follow-up courses e.g., computer vision, image processing, machine learning and data science.</td></tr>
<tr><td>Syllabus</td><td colspan="2">Linear Equations, Matrix Algebra, Vector Space, Eigenvalues and Eigenvectors , Orthogonality and Least Square, and Quadratic forms</td></tr>
</table>

Contents	***Topics**:* ***Chapter 1: Linear Equations in Linear Algebra:*** 1.1 *Systems of Linear Equations* 1.2 *Row Reduction and Echelon Forms* 1.3 *Vector Equations* 1.4 *The Matrix Equation Ax = b* 1.5 *Solution Sets of Linear Systems* 1.6 *Applications of Linear Systems* 1.7 *Linear Independence* 1.8 *Introduction to Linear Transformations* 1.9 *The Matrix of a Linear Transformation* *1.10 Linear Models in Business, Science, and Engineering* ***Chapter 2: Matrix Algebra:*** 2.1 *Matrix Operations* 2.2 *The Inverse of a Matrix* 2.3 *Characterizations of Invertible Matrices* 2.4 *Partitioned Matrices* 2.5 *Matrix Factorizations* 2.6 *Applications to Computer Graphics* 2.7 *Subspaces of* $\mathbb{R}^n$ 2.8 *Dimension and Rank* ***Chapter 3: Determinants:*** 3.1 *Introduction to Determinants* 3.2 *Properties of Determinants* 3.3 *Cramer's Rule* 3.4 *Volume* 3.5 *Linear Transformations* ***Chapter 4: Vector Spaces:*** 4.1 *Vector Spaces and Subspaces*

	4.2 *Null Spaces* 4.3 *Column Spaces, and Linear Transformations* 4.4 *Linearly Independent Sets; Bases, Coordinate Systems* 4.5 *The Dimension of a Vector Space* 4.6 *Rank* 4.7 *Change of Basis* ***Chapter 5: Eigenvalues and Eigenvectors****:* 5.1 *Eigenvectors and Eigenvalues* 5.2 *The Characteristic Equation* 5.3 *Diagonalization* 5.4 *Eigenvectors and Linear Transformations* 5.5 *Complex Eigenvalues* 5.6 *Discrete Dynamical Systems* ***Chapter 6: Orthogonality and Least Squares*** 6.1 *Inner Product, Length, and Orthogonality* 6.2 *Orthogonal Sets* 6.3 *Orthogonal Projections* 6.4 *The Gram–Schmidt Process* 6.5 *Least-Squares Problems* 6.6 *Applications to Linear Models* 6.7 *Inner Product Spaces* 6.8 *Applications of Inner Product Spaces* ***Chapter 7: Symmetric Matrices and Quadratic Forms****:* 7.1 *Diagonalization of Symmetric Matrices* 7.2 *Quadratic Forms* 7.3 *Constrained Optimization* 7.4 *The Singular Value Decomposition* 7.5 *Applications to Image Processing and Statistics* ***Chapter 8: The Geometry of Vector Spaces:***

<table>
<tr><td></td><td>8.1 Affine Combinations
8.2 Affine Independence
8.3 Convex Combinations
8.4 Hyper planes
Chapter 9: Optimization:
9.1 Matrix Games
9.2 Linear Programming—Geometric Method
9.3 Linear Programming—Simplex Method, Duality.</td></tr>
<tr><td>Assignments</td><td>Late submissions will not be accepted.
Assignments should be turned in at the start of the class.
Zero credit for turning in questions other than the assigned questions.</td></tr>
<tr><td>Assessment and Examinations</td><td><table>
<tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr>
<tr><td>1</td><td>Assignments
Quizzes
Class Participation</td><td>25%</td><td>There will be a graded quiz and assignments. The class participation will be framed so as to test the concepts involved in the lectures.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>It takes place at the end of the semester.</td></tr>
</table></td></tr>
<tr><td>Textbooks</td><td>1. Linear Algebra and Its Applications by David C. Lay, Steven R. Lay, Judi J. McDonald, 5th Edition, 2015, ISBN-13: 978-0321982384, ISBN-10: 032198238X</td></tr>
<tr><td>Reference Material/Suggested Readings</td><td>1. Introduction to Linear Algebra by Gilbert Strang, Fifth Edition, 2016, ISBN-13: 978-0980232776, ISBN-10: 0980232775</td></tr>
</table>

	2. *Elementary Linear Algebra by Howard Anton, 10th Edition, 2013, ISBN-13: 978-0470458211, ISBN-10: 0470458216* 3. *Coding the Matrix: Linear Algebra through Applications to Computer Science by Philip N. Klein, 1st Edition, 2013, ISBN-13: 978-0615880990, ISBN-10: 0615880991* 4. *Linear Algebra Labs with MATLAB by David Hill and David Zitarelli, 3rd Edition, 2003, ISBN-13: 978-0131432741, ISBN-10: 0131432745*

Detailed Lecture wise plan

Week No.	Lecture No.	Topic	Source Book-Chapter No. (Sections / Pages)	Recommendations for Learning Activities (Mention Assignments, Test, Quizzes, Practical, Case Study, Projects, Lab Work or Reading Assignments)
1	1	**Linear Equations in Linear Algebra** • Systems of Linear Equations • Row Reduction and Echelon Forms	Text. Ch1(1.1 to 1.2) Ref. 2 Ch1	Books Reading
	2	**Linear Equations in Linear Algebra** • Vector Equations • The Matrix Equation Ax = b	Text. Ch1(1.3 to 1.4) Ref. 1 Ch1 Ref. 2 Ch1	Books Reading
2	3	**Linear Equations in Linear Algebra** • Solution Sets of Linear Systems • Applications of Linear Systems	Text. Ch1(1.5 to 1.6) Ref. 1 Ch2 Ref. 2 Ch1	Books Readings
	4	**Linear Equations in Linear Algebra** • Solution Sets of Linear Systems • Applications of Linear Systems	Text. Ch1(1.5 to 1.6) Ref. 1 Ch2 Ref. 2 Ch1	Quiz#1 Books Readings
3	5	**Lab 1. Matrices in MATLAB** • Getting Data into MATLAB • Hilbert matrix • Dot product vs. cross product in MATALB	Ref. 4 Ch1 (1.1)	Books Readings
	6	**Linear Equations in Linear Algebra** • Linear Independence	Text. Ch1(1.7) Ref. 1 Ch3	
4	7	**Linear Equations in Linear Algebra** • Introduction to Linear Transformations	Text. Ch1(1.7) Ref. 1 Ch8 Ref. 2 Ch8	Books Readings

	8	**Linear Equations in Linear Algebra** • Introduction to Linear Transformations	Text. Ch1(1.8) Ref. 1 Ch8 Ref. 2 Ch8	Books Readings
5	9	**Lab 2. Linear System** • Row Operations Using MATLAB • Visualizing Row Operations • Symbolic Row Operations	Ref. 4 Ch1 (1.1)	Books Readings
	10	**Linear Equations in Linear Algebra** • Introduction to Linear Transformations • The Matrix of a Linear Transformation, Linear Models in Business, Science, and Engineering	Text. Ch1(1.8 to 1.9)	Assignment#1 Quiz#2 Books Readings
6	11	**Matrix Algebra** • Matrix Operations	Text. Ch2 (2.1)	Books Readings
	12	**Matrix Algebra** • The Inverse of a Matrix • Characterizations of Invertible Matrices	Text. Ch4 (2.2 to 2.3) Ref. 1 Ch1 Ref. 2 Ch1	Books Readings
7	13	**Matrix Algebra** • Partitioned Matrices	Text. Ch4 (2.4)	Books Readings
	14	**Matrix Algebra** • Matrix Factorizations	Text. Ch4 (2.5, 2.7)	Assignment#2 Quiz#3 Books Readings
8	15	**Matrix Algebra** • Applications to Computer Graphics	Text. Ch4 (2.5, 2.7)	Books Readings
	16	**Lab 3. Matrix Operations** Lab 4. Homogeneous Systems, Echelon Forms, and Inverses • Matrix Algebra • Generating Matrices • Display Formats • Homogeneous Systems • Reduced Row Echelon Form • Inverses	Ref.4 Lab 3 Sections (3.1, 3.2) Lab 4 Sections (4.1, 4.2, 4.3)	Books Readings

	MID TERM			
9	*17*	***Vector Spaces:*** • *Vector Spaces and Subspaces* • *Null Spaces* • *Column Spaces* • *Linear Transformations*	*Text. Ch4 (4.1 to 4.2)* *Ref. 2 Ch3* *Ref. 3 Ch3*	*Books Readings*
	18	*Midterm paper show*		
10	*19*	***Vector Spaces:*** • *Linearly Independent Sets* • *Bases* • *Coordinate Systems*	*Text. Ch4 (4.3 and 4.4)* *Ref. 2 Ch3* *Ref. 3 Ch3*	*Books Readings*
	20	***Vector Spaces:*** • *Paper Show* • *The Dimension of a Vector Space* • *Rank* • *Change of Basis*	*Text. Ch4 (4.5 to 4.7)* *Ref. 2 Ch3* *Ref. 3 Ch3*	*Books Readings*
11	*21*	***Eigenvalues and Eigenvectors**:* • *Eigenvectors and Eigenvalues* • *The Characteristic Equation* • *Diagonalization*	*Text. Ch5 (5.1 to 5.3)* *Ref. 1 Ch6* *Ref. 2 Ch3* *Ref. 3 Ch12*	*Books Readings* *Assignment#3* *Quiz#4*
	22	***Eigenvalues and Eigenvectors**:* • *Eigenvectors and Linear Transformations* • *Complex Eigenvalues* • *Discrete Dynamical Systems*	*Text. Ch5 (5.4 to 5.6)* *Ref. 1 Ch6* *Ref. 2 Ch3* *Ref. 3 Ch12*	*Books Readings*
12	*23*	***Orthogonality and Least Squares:*** • *Inner Product* • *Length and Orthogonality* • *Orthogonal Sets* • *Orthogonal Projections*	*Text. Ch6 (6.1 to 6.3)* *Ref. 1 Ch1* *Ref. 2 Ch6*	*Books Readings*

	24	***Orthogonality and Least Squares:*** • *The Gram–Schmidt Process* • *Least-Squares Problems* • *Applications to Linear Models* • *Inner Product Spaces* • *Applications of Inner Product Spaces*	*Text. Ch6 (6.4 to 6.8)* *Ref. 1 Ch4* *Ref. 2 Ch6*	*Books Readings*
13	*25*	***Symmetric Matrices and Quadratic Forms**:* • *Diagonalization of Symmetric Matrices* • *Quadratic Forms* • *Constrained Optimization*	*Text. Ch7 (7.1 to 7.3)* *Ref. 1 Ch6* *Ref. 2 Ch7*	*Books Readings* *Assignment#4* *Quiz#5*
	26	***Symmetric Matrices and Quadratic Forms**:* • *The Singular Value Decomposition* • *Applications to Image Processing and Statistics.*	*Text. Ch7 (7.4.to 7.5)* *Ref. 1 Ch7* *Ref. 2 Ch9* *Ref. 3 Ch11*	*Books Readings*
14	*27*	***Symmetric Matrices and Quadratic Forms**:* • *The Singular Value Decomposition* • *Applications to Image Processing and Statistics.*	*Text. Ch7 (7.4.to 7.5)* *Ref. 1 Ch7* *Ref. 2 Ch9* *Ref. 3 Ch11*	*Books Readings*
	28	***The Geometry of Vector Spaces:*** • *Affine Combinations* • *Affine Independence* • *Convex Combinations* • *Hyperplanes*	*Text. Ch8 (8.1.to 8.4)* *Ref. 2 Ch10*	*Books Readings*
15	*29*	***Optimization:*** • *Matrix Games,* • *Linear Programming—Geometric Method*	*Text. Ch9 (9.1.to 9.2)* *Ref. 2 Ch10* *Ref. 3 Ch13*	*Books Readings*
	30	***Optimization:*** • *Linear Programming—Simplex Method* • *Duality*	*Text. Ch9 (9.3.to 9.4)* *Ref. 3 Ch13*	*Books Readings*

16	*31*	***Lab 5. A Vector Space Example:*** • *Experimenting with Vector Space Properties* • *Linear Combinations* • *Span* • *Linear Independence/Dependence* • *Basis*	*Ref.4* *Lab 5 Sections (5.1)* *Lab 6 Sections (6.1 to 6.4)*	*Quiz#6*
	32	***Lab 6. Inner Product Spaces:*** • *The Standard Inner Product* • *Length and Distance* • *Orthogonal Bases* • *The Gram-Schmidt Process*	*Ref.4* *Lab 9 Sections (9.1, 9.2), Lab 10 Sections (10.1, 10.3)*	*Books Readings*
FINAL TERM				

Program	*BS Data Science*	
Course Code	*GE-190*	
Course Title	*Functional English*	
Credit Hours	***Theory***	***Lab***
	03	*0*
Lecture Duration	*90 minutes (1.5 Hours), 2 lectures per week*	
Semester	*1st*	
Pre-requisites	***Courses***	***Knowledge***
	None	*Before the commencement of this course students must possess the following skills:* • *At hand with the primary concepts of Grammar and its usage.* • *Basic understanding of Word Templates for Assignment documentation* • *Familiarization and practical experience of Microsoft Word, Microsoft PowerPoint*
Follow Up Courses	***Communication and Presentation Skills***	

Course Learning Outcomes (CLOs)

CLO No	***Course Learning Outcome***	***Bloom Taxonomy***
CLO-1	*Understand and use basic English grammar and vocabulary.*	*C-1*
CLO-2	*Write clear and concise emails and business correspondence.*	*C-2*
CLO-3	*Develop effective speaking and presentation skills for professional settings.*	*C-3*

Aims and Objectives	1. *The basic philosophy behind **English Composition & Comprehension** is to allow students to daily spend time writing*

	for real purposes about things that interest them. Students can experiment with a variety of writing genres. English, spelling, handwriting and other mechanics can be taught within ***English Composition & Comprehension****. Students learn the craft of writing through practice, conferring, and studying the craft of creative and fundamental writings.* 2. *English has opened for us several doors of knowledge for it is the lingua franca of the world and also the language of Science, Technology, Commerce and Diplomacy. The main objective of this course is to enhance English language skills of the students and develop their critical thinking.*
Learning Outcomes	• *Cultivating in the students the love for reading, and developing their oral and silent reading skills; also training students on critical reading and thinking.* • *Developing the students' four basic skills (listening, speaking, reading and writing).* • *Helping the students to expand their vocabulary and learn new vocabulary in context.* • *Introducing new grammatical structures and special difficulties and helping the students to understand and learn them.* • *Familiarizing the students with different writing styles and different text genres.* • *Developing the students' writing and paraphrasing skills through writing summaries and short compositions about the topics.*
Syllabus	*Parts of Speech, Grammar and types of Grammar; Sentence Errors, Paragraph and Essay Writing, Descriptive Essays Persuasive Essay,*

	Comparison and Contrast Essays, Narrative Essays, Dialogue Writing, Short Story Writing, Review Writing and Letter Writing.
Contents	*1. Language and its Basic Unit* *2. Sentence* *3. Grammar and types of Grammar* *4. Difference between British English and American English* *5. Parts of Speech:* *5.1. Noun* *5.2. Pronoun* *5.3. Adjective* *5.4. Verb* *5.5. Adverb* *5.6. Punctuation* *5.7. Conjunction* *5.8. Interjection* *6. Passage Comprehension* *7. Tenses and Translation* *8. Correction* *9. Change of Voice* *10. Change of Narration* *11. Paragraph writing* *12. Essay Writing* *12.1. Descriptive Essay* *12.2. Narrative Essay* *12.3. Comparison and contrast Essay* *12.4. Persuasive Essay* *13. Dialogue Writing* *14. Story Writing* *15. Application Writing*

<table>
<tr><td></td><td colspan="4">16. Letter Writing
17. Summary writing
18. Review Writing</td></tr>
<tr><td>Teaching-learning Strategies</td><td colspan="4">• Interactive class session
• Hands on practices of English Language Mechanisms in class
• Brainstorming and Group discussion sessions on topics</td></tr>
<tr><td>Assignments</td><td colspan="4">Paper based written assignments 8</td></tr>
<tr><td rowspan="4">Assessment and Examinations</td><td>Sr. #</td><td>Elements</td><td>Weightage</td><td>Details</td></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.</td></tr>
<tr><td>Textbooks</td><td colspan="4">1. College Writing Skills with Readings, by John Langan, McGraw-Hill, 5th Edition.</td></tr>
</table>

	2. A Textbook of English Prose and Structure by Arif Khattak, et al, GIKI Institute, 2000
Reference Material/Suggested Readings	*1. P. C. Wren & H. Martin "High School English Grammar & Composition"* *2. Colin W. Davis & Andrew J. Watts New Expressway For English 1 (New Edition)* *3. Diana Hacker. A Writer's Reference* *4. Sadat Ali Shah. Exploring The World Of English* *5. A. J. Thomson and A. V. Martinet. Practical English Grammar* *6. Handout provided by the teacher.* *7. Web Links: www.owl.english.purdue.edu*
Notes	• *Academic integrity is expected of all students.* • *Plagiarism or cheating in any assessment will result in at least an **F grade** in the course, and possibly more severe penalties.* • *Always be brief and to the point*

Detailed Lecture wise plan

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
1	1	• Introduction to the Subject • Language and its Basic Unit • Sentence • Sentence Structure • Types of Sentences • Lexicons	A(1) Handouts	• Distribution of course outline - • Worksheets
	2	• Grammar and types of Grammar	Handouts	Worksheets
2	3	• The Noun • Types of Nouns • Correction related to Noun	A(3)	Worksheets **Assignment # 1**
	4	• Pronoun • Types of Pronouns • Correction related to pronoun	A(4)	Worksheets
3	5	• Adjective • Types of Adjectives • Articles / Determiners etc. • Royal Oder and Order of Adjective • Correction Related to Adjective	A(5) Handouts	Worksheets
	6	• Verb • Types of Verbs • Correction Related to Verb • Auxiliary / Lexical verbs • Modal helping Verb	A(6) Handouts	Worksheets **Assignment # 2**
	7	• Adverb • Types of Adverbs • Correction related to Adverb	A(7)	**Quiz # 1**

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
4	8	• Punctuation • Types of Punctuation • Correction related to Punctuation	A(8)	Worksheets
5	9	• Preposition • Types of Prepositions • Correction Related to Preposition	A(9)	Worksheets **Assignment # 3**
	10	• Conjunction • Types of conjunction • Correction related to conjunction	A(10)	Worksheets
6	11	• Interjection • Types of Interjection • Correction related to Interjection	A(11)	Worksheets
	12	• Grammatical Terms • Gerunds,	A(12)	Worksheets **Quiz # 2**
7	13	• Phrases, clauses and sentences; linking phrases, transitions, coherence and unity.	A(13)	Worksheets
	14	• Phrases, clauses and sentences; linking phrases, transitions, coherence and unity.	A(14)	Worksheets
8	15	• Passage comprehension	B(9) Handouts	**Assignment # 4**
	16	• Change of Voice • Basic rules	Handouts	Worksheets
9	17	• Change of Voice	Handouts	Worksheets
	18	• Change of Narration • Basic Rules	Handouts	Worksheets
10	19	• Change of Narration • Interrogative sentences	Handouts	Worksheets
	20	• Change of Norriton • Exclamatory, Optative sentences	Handouts	Worksheets

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
11	21	• Paragraph Writing	B(10) Handouts	**Assignment # 5**
	22	• Descriptive Essay	B(11) Handouts	Class Task on the topic: Descriptive Essay
12	23	• Comparison and contrast Essay	B(12) Handouts	Class Task on the topic: Comparison and contrast Essay
	24	• Narrative Essay	B(13) Handouts	Class Task on the topic: Narrative Essay
13	25	• Persuasive Essay	B(14) Handouts	Class Task on the topic: Persuasive Essay **Assignment # 6**
	26	• Dialogue Writing	B(15)	Class Task on the topic: Dialogue Writing
14	27	• Application writing	B(5)	Class Task on the topic: Application writing **Quiz # 3**
	28	• Story writing	B(6)	Class Task on the topic: Story writing **Assignment # 7**
	29	○ Informal Letters	B(20) Handouts	Class Task on the topic: Informal Letters **Quiz # 4**

Week	***Lecture***	***Topic***	***Source Book (Ch#)***	***Recommendation for Learning Activities***
15	*30*	• *Summary Writing*	*B(3) Handouts*	*Class Task on the topic: Summary Writing* ***Assignment # 8***
16	*31*	• *Review Writing*	*Handouts*	*Class Task on the topic: Review Writing*
	32	• *Preparation of Final Exam*	*Revision*	*Revision / Best Wishes*

Program	*BS Data Science*	
Course Code	*GE-199*	
Course Title	*Expository Writing*	
Credit Hours	***Theory***	***Lab***
	3	*0*
Lecture Duration	*90 minutes (1.5 Hours), 2 lectures per week*	
Semester	*1*	
Pre-requisites	***Courses***	***Knowledge***
	English Composition & Comprehension	*Strong knowledge of English Grammar and Various Writing Mechanisms*
Follow Up Courses	*Technical and Business Writing*	
Aims and Objectives	1. *This course (**Communication and Presentation Skills**) is designed to provide students with the practical skills and knowledge of Communication necessary to express themselves clearly, with confidence and power, in a variety of writing and speaking situations.* 2. *Students will be taught presentation techniques; how to plan and structure an effective presentation; how to develop ideas; effective delivery methods; and how to overcome anxiety, fear and nervousness when making a presentation.*	
Learning Outcomes	• *Students will be able to document their data and sources according to the requirements of the business communication* • *To understand the essential points in preparing an oral presentation*	

	• *To understand the key elements of delivery of messages in oral presentations* • *To appreciate the nature of PowerPoint as a way of presenting the world* • *To develop more effective presentation skills* • *Parts of formal Letters and its layouts and standard layout*
Syllabus	*Principles of writing good English, Words, sentence and Paragraphs, Communication Components, Verbal and Non Verbal Communication, The Writing Process, 7Cs, Presentation, Memo, Business Letters, Minutes and Business Proposals*
Contents	*1. Principles of Writing good English (revision)* *2. What is Communication and Business Communication?* *3. Components of Communication* *4. The Writing Process:* *5. Basic Principles of Business Communications: 7Cs of Communications:* *5.1. Completeness* *5.2. Correctness* *5.3. Conciseness* *5.4. Courtesy* *5.5. Clarity* *5.6. Concreteness* *5.7. Consideration* *6. Verbal and Non-Verbal Communication* *7. Business Letter Writing* *8. Memo Writing* *9. Conducting Meetings and taking Minutes.* *10. Presentation skills*

<table>
<tr><td></td><td colspan="4">11. Use of Audio-Visual Aids
12. Job Interview
13. Reading Skills
14. Listening Skills</td></tr>
<tr><td>Teaching-learning Strategies</td><td colspan="4">Interactive class session
Hands on practices of English Language Mechanisms in class
Brainstorming and Group discussion sessions on topics
Expertise in professional messages
Presentation (Preparation and Practice)
Mock interviews</td></tr>
<tr><td>Assignments</td><td colspan="4">Paper based written assignments 8</td></tr>
<tr><td rowspan="4">Assessment and Examinations</td><td>Sr. #</td><td>Elements</td><td>Weightage</td><td>Details</td></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research</td></tr>
</table>

				proposal development, field work and report writing etc.
Textbooks	*1. Practical Business English, Collen Vawdrey, 1993, ISBN = 0256192740* *2. Effective Communication Skills: The Foundations for Change, John Nielsen, 2008, ISBN = 1453506748* *3. Various Web links*			
Reference Material/Suggested Readings	*1. Effective Business Communications 7th Edition by Herta A Murphy* *2. Business Communication Today, 14th edition by Courtland L Bovee and John Thill*			
Notes	• *Academic integrity is expected of all students.* • *Plagiarism or cheating in any assessment will result in at least an* ***F grade*** *in the course, and possibly more severe penalties.* • *Always be brief and to the point*			

Detailed Lecture wise plan

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
1	1	1. Overview of the Course Contents 2. Discussion of the Course 3. Revision of English Rules and Writings Mechanics		Distribution of course outline
	2	1. What is Communication? 2. What Is Business Communication? 3. What are different types of Communication	A(1)	
2	3	1. Communication 2.0 2. Communication 360° 3. Verbal Communication • Written Communication • Oral Communication	B(1) A(1)	**Assignment-1**
	4	• Non-Verbal communication 1. Effect of Non-Verbal Communication on: 1.1. Written Message 1.2. Physical Appearance 1.3. Surroundings	A(2)	Mock Exercise of Different Elements of Non Verbal Communication
3	5	2. Facial Expressions 3. Kinesics 4. Proxemics	A(2)	Mock Exercise of Different Elements of Non Verbal Communication
	6	5. Haptics 6. Time 7. Space 8. Paralinguistics	A(2)	Mock Exercise of Different Elements of Non Verbal Communication

Week	***Lecture***	***Topic***	***SourceBook (Ch#)***	***Recommendation for Learning Activities***
4	*7*	*The Writing Process* *1. Pre-Writing /Five Planning Steps* *1.1. Decide your Purpose* *1.2. Analyze/Anticipate your Audience* *1.3. Choose your ideas* *1.4. Select your Data* *1.5. Organize your Message*	*A(3)*	***Quiz # 1*** *Class practice of Pre Writing and its steps*
	8	*2. Drafting* *3. Editing* *4. Proofreading* *5. Revising* *6. Publishing*	*A(3)*	*Class practice of Drafting and Editing*
	9	*Basic Principles of Effective Communication Concept of 7Cs* *1. Completeness* *2. Correctness*	*A(4)*	*Worksheet* *Class practice of Basic Principles of Effective Communication*
5	*10*	*3. Conciseness* *4. Courtesy* *5. Clarity* *6. Concreteness* *7. Consideration*	*A(4)*	*Worksheet* *Class practice of Basic Principles of Effective Communication*

Week	***Lecture***	***Topic***	***SourceBook (Ch#)***	***Recommendation for Learning Activities***
6	*11*	*What is Memo?* *How to Write a Memo?* *What is Circular Message*	*A(5)*	***Assignment-2*** *Class practice of Memo Writing on different topics*
	12	*Parts of a Business Letter* *1. Standard Parts of a Business Letter*	*A(6)*	***Quiz # 2*** *Class practice of Memo Writing on different topics*
7	*13*	*2. Optional Parts of a Letter* *3. Format / Layout of a Business Letter*	*A(6)* *Handouts of Layout samples*	*Class practice of Memo Writing on different topics*
	14	*Types of Various Letters* *1. Inquiry Letter* *2. Reply of Inquiry Letter*	*A(7)* *Handouts of Sample text*	*Class practice of Memo Writing on different topics*
8	*15*	*3. Credit request Letter* *4. Reply of a credit Request Letter*	*A(7)* *Handouts of Sample text)*	***Assignment-3*** *Class practice of Memo Writing on different topics*
	16	*5. Order Letter* *6. New Orders* *7. Old Orders* *8. Acknowledgment Letter*	*A(7)*	***Assignment-4*** *Class practice of Memo Writing on different topics*

Week	***Lecture***	***Topic***	***SourceBook (Ch#)***	***Recommendation for Learning Activities***
	17	*9. Complaint Letter* *10. Reply of Complaint Letter*	*A(7)* *Hand outs*	***Assignment-5*** *Class practice of Memo Writing on different topics*
9	*18*	*11. Thank You Letter* *12. Dunning Letter*	*A(8)*	***Quiz# 3*** *Class practice of Memo Writing on different topics*
10	*19*	*What is Visual Aids?* *What is Visual Aids Media?*	*A(9)*	
	20	*Meeting Minutes* *How to take meeting minutes:* *1. Before meeting* *2. During meeting* *3. After meeting*	*A(10)*	***Assignment-6***
11	*21*	*What is Oral Presentation?* *Preparation of Oral Presentation:* *1. Before Presentation* *2. During Presentation* *3. After Presentation*	*A(11)*	
	22	*Methods of Oral Presentation* *1. Read from Manuscript* *2. Read form Memory*	*A(12)*	***Quiz# 4***
12	*23*	*3. Extemporaneous Speech* *4. Impromptu Speech*	*A(12)*	
	24	*Reading Skills* *Types of Readings*	*B(12)*	
13	*25*	*1. Listening Skills* *2. Types of Listening*	*B(12)*	
	26	*Barriers to listening* *How to overcome barriers to listening*	*B(13)*	

Week	**Lecture**	**Topic**	**SourceBook (Ch#)**	**Recommendation for Learning Activities**
14	27	*What are different Communication Barriers?* *How to overcome Communication Barriers?*	*B(13)*	***Assignment-7***
	28	*what is group?* *Types of groups?*	*A(20)*	
15	29	*What is meeting?* *Types of meetings*	*A(20)*	***Assignment-8***
	30	*What is job interview* *Types of job interview*	*A(21)*	*Mock Exercise*
16	31	*Preparation of interview* *Before interview* *During interview* *After interview*	*A(21)*	*Mock Exercise*
	3	*Revision*	*Hand outs*	

<table>
<tr><td>Program</td><td colspan="2">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="2">MS-254</td></tr>
<tr><td>Course Title</td><td colspan="2">Technical and Business Writing</td></tr>
<tr><td rowspan="2">Credit Hours</td><td>Theory</td><td>Lab</td></tr>
<tr><td>03</td><td>00</td></tr>
<tr><td>Lecture Duration</td><td colspan="2">90 minutes (1.5 Hours), 2 lectures per week</td></tr>
<tr><td>Semester</td><td colspan="2">7th</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td>Courses</td><td>Knowledge</td></tr>
<tr><td>Communication and Presentation Skills</td><td>Before the commencement of this course students must possess following skills:
• At hand with the primary concepts of communication and its process in the business world.
• Basics of Business Writing Mechanics
• Preparation of Professional Documents
• Fundamental knowledge of Tenses and English structure of sentences
• Basic understanding of MS Word Templates for business documentation
•</td></tr>
<tr><td>Follow Up Courses</td><td colspan="2">No Any</td></tr>
<tr><td>Aims and Objectives</td><td colspan="2">If you effectively complete readings, practice exercises, workshops and assignments this course should:</td></tr>
</table>

	1. provide you with the confidence to use written communication in your work and personal experience in and beyond college. 2. acquaint you with the concept of a writer-reader relationship and identify the need for active participation from both writer and reader. 3. teach you the skills needed to successfully communicate in a modern world through written materials. 4. learn to identify and select many **types of writing** frequently required in a variety of careers. 5. determine your purposes/objectives and develop skill in **composing and revising on the computer** documents with formats and language appropriate for those purposes. 6. demonstrate in your writing the **effective communication principles** encouraged by professional writers. 7. achieve a greater awareness of the importance of selecting and integrating **graphics** with written communication.
Learning Outcomes	**The core of this course can be summarized in the following four skills area:** • Research Skills (using primary and library research to discover and employ information) • Correspondence Skills (learning the generic conventions of each) • Promotional Writing Skills (may or may not use primary research; to disseminate information; to inform and persuade public audiences that organizations communicate with) • Visual Communication Skills (may appear as separate assignments or as components of other assignments)
Syllabus	Organizing information and generation solution: brainstorming, organizing material, construction of the formal outline. Document design: document structure, preamble, summaries, Abstracts, table

	of contents, footnotes, glossaries, cross referencing, plagiarism, citation and bibliography, glossaries, index, appendices, typesetting systems Overview of technical reporting, Leaflets, brochures, handbooks and magazines. Visual Communication and Visual Aids Media. Business Communication and Ethical Issues: Business Communication and Technological Context: Business Communication and Legal Issues: Individual and National Cultural Variables ; Job Interview, Job Application, Different types of Reports and Proposals
Contents	1. Overview of technical reporting, use of library and information gathering, administering questionnaires, reviewing the gathered information. 2. Organizing information and generation solution: brainstorming, organizing material, construction of the formal outline. document design: document structure, preamble, summaries, 3. Abstracts, table of contents, footnote, glossaries, cross referencing, plagiarism, citation and bibliography, glossaries, index, appendices, typesetting systems. 4. Visual Communication and Visual Aids Media 5. Business Communication and Ethical Issues 6. Business Communication and Technological Context 7. Business Communication and Legal Issues 8. Individual and National Cultural Variables 9. Job Interview 10. Job Application 11. **Follow up interview** 12. Different types of Reports

	1. Progress Reports 2. Conference Report 3. Informational Reports. 4. Analyzing a Case 5. Writing a Case Report 6. Discussing a Case Study 7. Presenting a Case Study 13. Leaflets, brochures, handbooks, magazines, articles, research papers, feasibility reports, project reports, technical research reports,
Teaching-learning Strategies	• Interactive class session • Hands on practices in class • Brainstorming and Group discussion sessions
Assignments	Paper based written assignments 8 Class tasks
Assessment and Examinations	

Sr. #	Elements	Weightage	Details
1	Formative Assessment	25%	It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.
2	Midterm Assessment	35%	It takes place at the mid-point of the semester.

	3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.
Textbooks	1. Technical Report Writing, by Pauley and Riordan, Houghton Mifflin Company, 8th Edition. 2. Effective Technical Communication by Ashraf Rizvi, Tata McGraw-Hill.			
Reference Material/Suggested Readings	4. Effective Business Communications, 7th Edition by Herta A Murphy 5. Business Communication Today, 14th edition by Courtland L Bovee and John Thill			
Notes	• Academic integrity is expected of all students. • Plagiarism or cheating in any assessment will result in at least an F grade in the course, and possibly more severe penalties. • Be Brief and to the point			

Detailed Lecture wise plan

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
1	1	Introduction to the Subject		Distribution of Course outline
	2	Concepts of Communication (Revision)		
2	3	Overview of technical reporting, use of library and information gathering, administering questionnaires, reviewing the gathered information.	A(17)	**Assignment # 1**
	4	Overview of Leaflets, brochures, handbooks, magazines, articles, research papers	A(17)	Class Task
3	5	Abstracts, table of contents, footnotes, glossaries, cross referencing, plagiarism, citation and bibliography, glossaries, index, appendices, typesetting systems.	A(17)	**Quiz # 1** Class Task
	6	Analyzing a Case	B(7) Handouts	Class Task
4	7	Writing a Case Report	B(7) Handouts	**Assignment # 2**
	8	Discussing a Case Study	A(18) Handouts	Class Task
5	9	Presenting a Case Study	A(18) Handouts	Class Task
	10	A formal Case and its written analysis	A(18) Handouts	**Quiz # 2**

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
6	11	Conversion of a written report into an oral presentation	A(18) Handouts	Class Task
	12	What is Report? Types of report Short report Long report	B(18)	
7	13	Parts of a report	A(19) Handouts	Class Task
	14	Conference report	A(19) B(8) Handouts	**Assignment # 3**
8	15	Analytical report	A(19) B(8)	**Assignment # 4** Class Task
	16	Informational report	Handouts	Class Task
9	17	Progress report	B(15) A(20) Handouts	Class Task
	18	Feasibility Report	A(19) B(10)	**Assignment # 5**
10	19	What is RFP? What is business Proposal	A(18) B(15) Handouts	**Quiz # 3**
	20	Business Proposal	A(20) B(15) Handouts	Class Task
11	21	Sales Proposal	B(14) Handouts	**Assignment # 6**
	22	Visual Communication and Visual Aids Media	B(15)	**Quiz # 4**
12	23	Business Communication and Ethical Issues	B(12)	
	24	Business Communication and Technological Context	B(13)	
13	25	Business Communication and Legal Issues	B(14)	
	26	Individual and National Cultural Variables	B(16)	

Week	**Lecture**	**Topic**	**Source Book (Ch#)**	**Recommendation for Learning Activities**
14	27	Job application letters	A(14)	**Assignment # 7** Class Task
	28	Types of Résumés	A(20)	
15	29	Résumés	A(20)	**Assignment # 8** Class Task
	30	Job Interviews	A(20)	Mock Exercise
16	31	Follow up on interviews	A(20)	Mock Exercise
	32	Revision	Hand outs	

<table>
<tr><td>Program</td><td colspan="2">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="2">GE-168</td></tr>
<tr><td>Course Title</td><td colspan="2">Ideology and Constitution of Pakistan</td></tr>
<tr><td rowspan="2">Credit Hours</td><td>Theory</td><td>Lab</td></tr>
<tr><td>2</td><td>0</td></tr>
<tr><td>Lecture Duration</td><td colspan="2">60 minutes (1 Hours), 2 lectures per week</td></tr>
<tr><td>Semester</td><td colspan="2">1</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td>Courses</td><td>Knowledge</td></tr>
<tr><td></td><td></td></tr>
<tr><td>Follow Up Courses</td><td colspan="2"></td></tr>
<tr><td>Aims and Objectives</td><td colspan="2">1. To teach the students about the objectives, freedom and various governments in Pakistan, along with its leaders, culture, geography and other aspects.</td></tr>
<tr><td>Learning Outcomes:</td><td colspan="2">Students get comprehensive know how of History, Geography, Politics, Leaders and different aspects related to our homeland Pakistan.</td></tr>
<tr><td>Syllabus</td><td colspan="2">Historical background of Pakistan: Muslim society in Indo-Pakistan, the movement led by the societies, the downfall of Islamic society, the establishment of British Raj- Causes and consequences. Political evolution of Muslims in the twentieth century: Sir Syed Ahmed Khan; Muslim League; Nehru; Allama Iqbal: Independence Movement; Lahore Resolution; Pakistan culture and society, Constitutional and Administrative issues, Pakistan and its geo-political dimension, Pakistan and International Affairs, Pakistan and the challenges ahead.</td></tr>
<tr><td>Contents</td><td colspan="2">1. Historical background of Pakistan:
2. Muslim society in Indo-Pakistan, the movement led by the societies, the downfall of Islamic society, the establishment of British Raj- Causes and consequences.</td></tr>
</table>

	3. Political evolution of Muslims in the twentieth century: Sir Syed Ahmed Khan; 4. Muslim League; Nehru; Allama Iqbal: 5. Independence Movement; Lahore Resolution; 6. Pakistan culture and society, 7. Constitutional and Administrative issues, Pakistan and its geo-political dimension, Pakistan and International Affairs, Pakistan and the challenges ahead.
Teaching-learning Strategies	The students will be given all round knowledge of the subject oral/on white board/assignments/sudden quizzes during class rooms only.
Assignments	Assignments will be assigned throughout the course.

Assessment and Examinations

Sr. #	Elements	Weightage	Details
1	Formative Assessment	25%	It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.
2	Midterm Assessment	35%	It takes place at the mid-point of the semester.
3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.

Textbooks	1. The Emergence of Pakistan, Chaudary M., 1967 2. The making of Pakistan, Aziz. 1976 3. A Short History of Pakistan, I. H. Qureshi, ed., Karachi, 1988

Reference Material/Suggested Readings	All reference Material/readings will be provided during lectures as per the class performance and their interest in the degree program overall.
Notes	Students will take their own notes during class.

Program	BS Data Science	
Course Code	GE-163	
Course Title	Islamic Studies	
Credit Hours	**Theory**	**Lab**
	02	0
Lecture Duration	60 minutes (1 Hours), 2 lectures per week	
Semester	2	
Pre-requisites	**Courses**	**Knowledge**
Follow Up Courses		

Course Learning Outcomes (CLOs)

CLO No	**Course Learning Outcome**	**Bloom Taxonomy**
CLO-1	To further enhance the knowledge of Islam.	C3 (Apply)
CLO-2	To understand the basic concept of Islam and Quran Pak.	C2 (Understand)
CLO-3	To understand the concept of Haqooq ul ibad in the light of Quran.	C2 (Understand)
CLO-4	To know the importance of Islamic concept about other religions.	C2 (Understand)

Aims and Objectives	1. To teach students about Islam
Learning Outcomes:	Students will learn 1. Basic Themes of Quran, 2. Introduction to Sciences of Hadith, 3. Introduction to Islamic Jurisprudence, 4. Primary & Secondary Sources of Islamic Law, 5. Makken & Madnian life of the Prophet, 6. Islamic Economic System,

	7. Political theories, 8. Social System of Islam
Syllabus	Basic Themes of Quran, Introduction to Sciences of Hadith, Introduction to Islamic Jurisprudence, Primary & Secondary Sources of Islamic Law, Makken & Madnian life of the Prophet, Islamic Economic System, Political theories, Social System of Islam
Contents	1. Basic Themes of Quran, 2. Introduction to Sciences of Hadith, 3. Introduction to Islamic Jurisprudence, 4. Primary & Secondary Sources of Islamic Law, 5. Makken & Madnian life of the Prophet, 6. Islamic Economic System, 7. Political theories, 8. Social System of Islam
Teaching-learning Strategies	The students will be given all round knowledge of the subject oral/on white board/assignments/sudden quizzes during class rooms only.
Assignments	Assignments will be assigned throughout the course.

Assessment and Examinations

Sr. #	Elements	Weightage	Details
1	Formative Assessment	25%	It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.
2	Midterm Assessment	35%	It takes place at the mid-point of the semester.
3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their

				students based on term paper, research proposal development, field work and report writing etc.
Textbooks	1. Introduction to Islam by Dr Hamidullah, Papular Library Publishers Lahore 2. Principles of Islamic Jurisprudence by Ahmad Hassan, Islamic Research Institute, IIUI 3. Muslim Jurisprudence and the Quranic Law of Crimes, By Mir Waliullah, Islamic Books Services			
Reference Material/Suggested Readings	All reference Material/readings will be provided during lectures as per the class performance and their interest in the degree program overall.			
Notes	Students will take their own notes during class.			

Program	BS Data Science	
Course Code	GE-362	
Course Title	Entrepreneurship	
Credit Hours	**Theory**	**Lab**
	03	0
Lecture Duration	90 minutes (1.5 Hours), 2 lectures per week	
Semester		
Pre-requisites	**Courses**	**Knowledge**
Follow Up Courses		
Aims and Objectives	1. To teach students about fundamentals of Entrepreneurship	
Learning Outcomes:	Students will the fundamentals of Entrepreneurship	
Syllabus	Intro to entrepreneurship, Historical perspective of entrepreneurship, The four dimensions of venture creation, Competing models of entrepreneurship, Effectual vs. causal logic of entrepreneurship, Entrepreneurial thinking versus managerial thinking, Effectuation model of entrepreneurship, Principles of effectual entrepreneurship, Idea vs. opportunity, Essential qualities of an opportunity, Ways to 'find' a business opportunity, Window of opportunity, Role of feasibility analysis in developing successful business ideas, Characteristics of attractive industries for start---ups, Financial and commercial merit of the business idea, The venture opportunity profile, Opportunities found vs. Opportunities created, The model of opportunity 'creation', The 'curry in a hurry' principle, Value Innovation: A new mantra of value creation, The Strategy Canvas, Four---actions framework, Business Idea Presentations: Groups will present and	

<table>
<tr><td></td><td>submit their Business Concept Statements according to the format prescribed by the instructor.</td></tr>
<tr><td>Contents</td><td>1. Intro to entrepreneurship,
2. Historical perspective of entrepreneurship,
3. The four dimensions of venture creation, Competing models of entrepreneurship,
4. Effectual vs. causal logic of entrepreneurship, Entrepreneurial thinking versus managerial thinking, Effectuation model of entrepreneurship,
5. Principles of effectual entrepreneurship, Idea vs. opportunity, Essential qualities of an opportunity, Ways to ‘find’ a business opportunity, Window of opportunity,
6. Role of feasibility analysis in developing successful business ideas, Characteristics of attractive industries for start---ups, Financial and commercial merit of the business idea,
7. The venture opportunity profile, Opportunities found vs. Opportunities created, The model of opportunity ‘creation’, The ‘curry in a hurry’ principle, Value Innovation: A new mantra of value creation,
8. The Strategy Canvas, Four---actions framework, Business Idea
9. Presentations: Groups will present and submit their Business Concept
10. Statements according to the format prescribed by the instructor.</td></tr>
<tr><td>Teaching-learning Strategies</td><td>The students will be given all round knowledge of the subject oral/on white board/assignments/sudden quizzes during class rooms only.</td></tr>
<tr><td>Assignments</td><td>Assignments will be assigned throughout the course.</td></tr>
<tr><td>Assessment and Examinations</td><td>
<table>
<tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
</table>
</td></tr>
</table>

	3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.
Textbooks	• B. R. Barringer, and R. D. Ireland, Entrepreneurship: Successfully Launching New Ventures, 3rd Edition, Prentice Hall, 2009, ISBN: 0138158088. • J. Timmons, S. Spinelli, New Venture Creation – Entrepreneurship for 21st Century, 8th Edition, McGraw-Hill, 2008, ISBN: 0071276327.			
Reference Material/Suggested Readings	All reference Material/readings will be provided during lectures as per the class performance and their interest in the degree program overall.			
Notes	Students will take their own notes during class.			

Program	BS Data Science		
Course Code	GE-402		
Course Title	Professional Practices		
Credit Hours	**Theory**	**Lab**	
	03	0	
Lecture Duration	90 minutes (1.5 Hours), 2 lectures per week		
Semester			
Pre-requisites	**Courses**	**Knowledge**	
Follow Up Courses			
Course Learning Outcomes (CLOs)			
CLO No	**Course Learning Outcome**	**Bloom Taxonomy**	
CLO-1	Identify the content of religious, national, or international law dealing with professional ethics	A-2	
CLO-2	Apply the knowledge of ethics in their personal and professional life.	A-3	
CLO-3	Gain the ability to enhance key factors of interpersonal relations, to follow and implement the acquired knowledge of ethical skills in given situations by controlling his/her temperament.	A-4	
Aims and Objectives	1. To teach students various Professional Practices.		
Learning Outcomes:	Students will various Professional Practices.		
Syllabus	Computing Profession, Computing Ethics, Philosophy of Ethics. The Structure of Organizations, Finance and Accounting, Anatomy of a Software House, Computer Contracts, Intellectual Property Rights, The		

<table>
<tr><td></td><td colspan="4">Framework of Employee Relations Law and Changing Management Practices, Human Resource Management and IT, Health and Safety at Work, Software Liability, Liability and Practice, Computer Misuse and the Criminal Law, Regulation and Control of Personal Information. Overview of the British Computer Society Code of Conduct, IEEE Code of Ethics, ACM Code of Ethics and Professional Conduct, ACM/IEEE Software Engineering Code of Ethics and Professional Practice. Accountability and Auditing, Social Application of Ethics.</td></tr>
<tr><td>Contents</td><td colspan="4">1. Computing Profession, Computing Ethics, Philosophy of Ethics. The Structure of Organizations,
2. Finance and Accounting, Anatomy of a Software House,
3. Computer Contracts, Intellectual Property Rights, The Framework of Employee Relations Law and Changing Management Practices,
4. Human Resource Management and IT, Health and Safety at Work, Software Liability, Liability and Practice,
5. Computer Misuse and the
6. Criminal Law, Regulation and Control of Personal Information. Overview of the British Computer Society Code of Conduct,
7. IEEE Code of Ethics, ACM Code of Ethics and Professional Conduct, ACM/IEEE Software Engineering Code of Ethics and Professional Practice. Accountability and Auditing, Social Application of Ethics.</td></tr>
<tr><td>Teaching-learning Strategies</td><td colspan="4">The students will be given all round knowledge of the subject oral/on white board/assignments/sudden quizzes during class rooms only.</td></tr>
<tr><td>Assignments</td><td colspan="4">Assignments will be assigned throughout the course.</td></tr>
<tr><td rowspan="3">Assessment and Examinations</td><td>Sr. #</td><td>Elements</td><td>Weightage</td><td>Details</td></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
</table>

	3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.
Textbooks	• Professional Issues in Software Engineering by Frank Bott, Allison Coleman, Jack Eaton and Diane Rowland, CRC Press; 3rd Edition (2000). ISBN-10: 0748409513 • Computer Ethics by Deborah G. Johnson, Pearson; 4th Edition (January 3, 2009). ISBN-10: 0131112414 • A Gift of Fire: Social, Legal, and Ethical Issues for Computing and the Internet (3rd Edition) by Sara Baase, Prentice Hall; 3rd Edition (2008). ISBN-10: 0136008488 • Applied Professional Ethics by Gregory R. Beabout, University Press of America (1993). ISBN-10: 0819193747.			
Reference Material/Suggested Readings	All reference Material/readings will be provided during lectures as per the class performance and their interest in the degree program overall.			
Notes	Students will take their own notes during class.			

Program	BS Data Science	
Course Code	ED-333	
Course Title	Theory of Automata and Formal Languages	
Credit Hours	**Theory**	**Lab**
	03	0
Lecture Duration	90 minutes (1.5 Hours), 2 lectures per week	
Semester	8	
Pre-requisites	**Courses**	**Knowledge**
	Nil	
Follow Up Courses	Compiler Construction, Concepts and comparison of PLs	

Course Learning Outcomes (CLOs)

CLO No	**Course Learning Outcome**	**Bloom Taxonomy**
CLO-1	Explain and manipulate the different concepts in automata theory and formal languages such as formal proofs, automata, regular expressions, Turing machines etc.	C2 (Understand)
CLO-2	Prove properties of languages, grammars and automata with rigorously formal mathematical methods.	C2 (Understand)
CLO-3	Design of automata, RE and CFG.	C6 (Design)
CLO-4	Transform between equivalent NFAs, DFAs and Res.	C3 (Apply)
CLO-5	Define Turing machines performing simple tasks.	C2 (Understand)
CLO-6	Differentiate and manipulate formal descriptions of languages, automata and grammars with focus on regular and context-free languages, finite automata and regular expressions.	C3 (Apply)

Aims and Objectives	The following are instructional aims and objectives of the course; they are by no means complete, but they characterize the kind of understanding that I hope students will develop over the quarter. At the end of the course students should: 1. Be able to prepare/analyze Programming Language specifications. 2. Be able to work in the areas like Language design and Natural Language Processing etc. 3. Be able to decide what sort of rules may be included in Programming Languages and how these rules are enforced by language processors like compilers and interpreters.
Learning Outcomes:	A student who has passed this course shall be able to 1. Understand and manipulate formal descriptions of languages, automata and grammars with focus on Regular and Context Free Languages, Finite State Automata and Regular Expressions. 2. Apply rigorously formal mathematical methods to prove properties of languages, grammars and automata. 3. Apply the knowledge for designing and implementing the Programming Languages.
Syllabus	This course introduces the fundamentals of Programming Language Theory, Regular Expressions, and Grammars. Deterministic and non-deterministic automata, their limitations and alternates like pushdown automata. Different computing models are also discussed such as Automata and Turing Machines with emphasis on their applicability to practical problem domains.
Contents	15. Fundamentals of Programming Language Theory 16. Regular Expressions 17. Grammars 18. Deterministic and non-deterministic automata 19. Their limitations and alternates like pushdown automata. 20. Different computing models 21. Automata

	22. Turing Machines 23. Emphasis on their applicability to practical problem domains.			
Teaching-learning Strategies	The students will be given all round knowledge of the subject oral/on white board/assignments/sudden quizzes during class rooms only.			
Assignments	Assignments will be assigned throughout the course.			
Assessment and Examinations	Sr. #	Elements	Weightage	Details
	1	Formative Assessment	25%	It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.
	2	Midterm Assessment	35%	It takes place at the mid-point of the semester.
	3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.
Textbooks	Introduction to Computer Theory, by Daniel I. Cohen, John Wiley & Sons			
Reference Material/Suggested Readings	All reference Material/readings will be provided during lectures as per the class performance and their interest in the degree program overall.			

Notes	Students will take their own notes during class.

Detailed Lecture wise plan

Unit	Sub-Topic	Readings	Activities
1	Introduction of class and subject		
2	Language, string, valid string, character set	Chapter # 1	
3	Definitions of various language related terms, Recursive Definition	Chapter # 2,3	Yes
4	Regular expressions, Regular languages	Chapter # 4	Yes
5			
6	Finite Automata and its types, Transition Graph, Generalized Transition Graph, F.A. with output	Chapter # 5,6,8	Yes
7			
8	Kleene's Theorem, Regular Languages	Chapter # 7,10	Yes
9			
10	Context Free Grammars	Chapter # 13,14	
11	Various grammatical formats	Chapter # 15,16	Yes
12	Push Down Automata and its types	Chapter # 17	
13	PDA with N-stacks/Post Machine	Chapter # 25	
14	Turing Machine	Chapter # 24	Yes
15	The Chomsky Hierarchy, Review and Revision	Chapter # 30	

<table>
<tr><td>Program</td><td colspan="3">BS data science</td></tr>
<tr><td>Course Code</td><td colspan="3">ED-343</td></tr>
<tr><td>Course Title</td><td colspan="3">Advance Database Management Systems</td></tr>
<tr><td rowspan="2">Credit Hours</td><td colspan="2">Theory</td><td>Lab</td></tr>
<tr><td colspan="2">3</td><td>0</td></tr>
<tr><td>Lecture Duration</td><td colspan="3">90 minutes (1.5 Hours), 2 lectures per week</td></tr>
<tr><td>Semester</td><td colspan="3">7</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td colspan="2">Courses</td><td>Knowledge</td></tr>
<tr><td colspan="2">Database Systems</td><td>Nil</td></tr>
<tr><td>Follow Up Courses</td><td colspan="3"></td></tr>
<tr><td colspan="4">Course Learning Outcomes (CLOs)</td></tr>
<tr><td>CLO No</td><td colspan="2">Course Learning Outcome</td><td>Bloom Taxonomy</td></tr>
<tr><td>CLO-1</td><td colspan="2">Understanding advance data models, technologies and approaches for building distributed database systems.</td><td>C2 (Understand)</td></tr>
<tr><td>CLO-2</td><td colspan="2">Applying the models and approaches in order to become enabled to select and apply appropriate methods for a particular case.</td><td>C3 (Apply)</td></tr>
<tr><td>CLO-3</td><td colspan="2">To develop a database solution for a given scenario/ challenging problem in the domain of distributed database systems.</td><td>C3 (Apply)</td></tr>
<tr><td>Aims and Objectives</td><td colspan="3">Upon completion of this course, the student will be able to:
1. Grasp theory and science of database systems
2. Design distributed databases
3. Write structured queries</td></tr>
<tr><td>Learning Outcomes</td><td colspan="3">At the end of the course, you should be able to:
• Understanding advance data models, technologies and approaches for building distributed database systems
• Applying the models and approaches in order to become enabled to select and apply appropriate methods for a particular case</td></tr>
</table>

	• To develop a database solution for a given scenario/ challenging problem in the domain of distributed database systems.
Contents	1. Background 1.1. Introduction to advance data models 1.2. Object relational introduction 1.3. Object oriented introduction 1.4. File organization 2. Transaction and batch processing 2.1. Transaction 2.2. Concurrency 2.3. Recovery and backup 3. Database Management System 3.1. Transactional processing 3.2. Concurrency control techniques 3.3. Recovery techniques 3.4. Query processing 3.5. Query optimization 4. Database Programming 4.1. PL/SQL 4.2. T-SQL 4.3. Similar technology) 5. Integrity and security 5.1. Integrity 5.2. Security 6. Database Administration 6.1. Role management 6.2. Managing database access 6.3. Views 7. Physical database design and tuning

<table>
<tr><td></td><td>7.1. Distributed database systems
8. Emerging research trends in database systems,
8.1. MONGO DB
8.2. NO SQL
8.3. Or similar technologies</td></tr>
<tr><td>Teaching-learning Strategies</td><td>• Interactive class session
• Hands on practices in class
• Brainstorming and Group discussion sessions</td></tr>
<tr><td>Assignments</td><td>• Paper based written assignments 3
• Project 2
• Quiz 4</td></tr>
<tr><td>Assessment and Examinations</td><td>
<table>
<tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper.</td></tr>
</table>
</td></tr>
<tr><td>Textbooks & Reference material</td><td>• Database Systems: A Practical Approach to Design, Implementation, and Management, 6th Edition by Thomas Connolly and Carolyn Begg
• 2. Database Management Systems, 3rd Edition by Raghu Ramakrishnan, Johannes Gehrke
• 3. Database System Concepts, 6th Edition by Avi Silberschatz, Henry F. Korth and S. Sudarshan.</td></tr>
</table>

	• 4. Database Systems: The Complete Book, 2nd Edition by Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer Widom
Notes	• The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester.

<table>
<tr><td>Program</td><td colspan="2">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="2">ED-344</td></tr>
<tr><td>Course Title</td><td colspan="2">Machine Learning</td></tr>
<tr><td rowspan="2">Credit Hours</td><td>Theory</td><td>Lab</td></tr>
<tr><td>2</td><td>1</td></tr>
<tr><td>Lecture Duration</td><td colspan="2">60 minutes (1Hours), 2 lectures per week, 3 hours lab session per week</td></tr>
<tr><td>Semester</td><td colspan="2">6</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td>Courses</td><td>Knowledge</td></tr>
<tr><td>Nil</td><td>Basic experience with programming is required. Knowledge of Linear Algebra, Probability and Statistics, and Calculus would be helpful.</td></tr>
<tr><td>Follow Up Courses</td><td colspan="2">Deep Learning</td></tr>
</table>

Course Learning Outcomes (CLOs)

CLO No	**Course Learning Outcome**	**Bloom Taxonomy**
CLO-1	Describe basic machine learning concepts, theories and applications.	C1 (Knowledge)
CLO-2	Apply supervised learning techniques to solve classification problems of moderate complexity.	C3 (Apply)
CLO-3	Apply unsupervised learning techniques to solve clustering problems of moderate complexity.	C3 (Apply)
CLO-4	Apply reinforcement learning algorithms to environments with complex dynamics.	C3 (Apply)
CLO-5	Develop a reasonable size project using suitable machine learning technique to solve complex problems.	C6 (Create)

Aims and Objectives	1. This course aims to introduce students to the basics of machine learning 2. The general objective is to make students understand a range of various machine learning algorithms along with their strengths and weaknesses. 3. The course aims to make students able to apply machine learning algorithms to solve problems in Data Science of moderate complexity.
Learning Outcomes	• Students should be able to describe basic machine learning concepts, theories and applications • Students should be able to apply supervised/unsupervised learning techniques to solve classification/clustering problems of moderate complexity • Students should be able to apply reinforcement learning algorithms to environments with complex dynamics • Students should be able to develop a reasonable size project using suitable machine learning technique
Syllabus	Introduction to machine learning; concept learning: General-to-specific ordering of hypotheses, Version spaces, Candidate elimination algorithm; Supervised Learning: Decision trees, Naive Bayes, Artificial Neural Networks, Support Vector Machines, Overfitting, noisy data, and pruning, Measuring Classifier Accuracy; Linear and Logistic regression; Unsupervised Learning: Hierarchical Agglomerative Clustering. k-means partitional clustering; Self-Organizing Maps (SOM) k-Nearest-neighbor algorithm; Semi-supervised learning with EM using labeled and unlabeled data; Reinforcement Learning: Hidden Markov models, Monte Carlo inference Exploration vs. Exploitation Trade-off, Markov Decision

	Processes; Ensemble Learning: Using committees of multiple hypotheses. Bagging, boosting.
Contents	Unit 1: Introduction 1.1 Machine learning 1.2 Concept learning 1.3 General to specific ordering of hypotheses 1.4 Introduction to version space 1.5 Candidate elimination algorithm Unit 2: Supervised Learning 2.1 Introduction 2.2 Decision Trees 2.3 Naïve Bayes and its types 2.4 Artificial Neural Networks 2.5 Support Vector Machines 2.6 Linear and Logistic regression Unit 3: Unsupervised Learning 3.1 Introduction 3.2 Partition and hierarchical clustering methods 3.3 Self-Organizing maps Unit 4: Semi-supervised Learning 4.1 Introduction 4.2 Expectation maximization (EM) Unit 5: Reinforcement Learning 5.1 Introduction 5.2 Hidden Markov Models 5.3 Monte Carlo inference

<table>
<tr><td></td><td>5.4 Exploration vs exploitation trade-off, MDP
Unit 6: Ensemble Learning
6.1 Introduction
6.2 Using committees of multiple hypotheses
6.3 Bagging, Boosting</td></tr>
<tr><td>Teaching-learning Strategies</td><td>Multimedia presentations involving interaction from students
Hands on exercises for concept reinforcement
Coding in laboratory</td></tr>
<tr><td>Assignments</td><td>There would be 4-5 programming assignments (2 pre and 2-3 post midterm)</td></tr>
<tr><td>Assessment and Examinations</td><td>
<table>
<tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their</td></tr>
</table>
</td></tr>
</table>

				students based on term paper, research proposal development, field work and report writing etc.
Textbooks	Mitchell, T. M. (1997). Machine Learning. (1st Edition). McGraw-Hill Education. ISBN-13: 978-0070428072			
Reference Material/Suggested Readings	• Murphy, K. P. (2012). Machine Learning: A Probabilistic Perspective. MIT Press. ISBN-13: 978-0262018029 • Bishop, C. M. (2006). Pattern Recognition and Machine Learning. New York: Springer-Verlag. ISBN-13: 978-0387310732			
Notes	• Academic integrity is expected of all students. Plagiarism or cheating in any assessment will result in at least an F grade in the course, and possibly more severe penalties • There is no makeup for a missed sessional grading instruments like quizzes, assignments, and homework • The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester			

Detailed Lecture wise plan

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
1	1	Introduction: types; well-posed ML problems	Ch#1	Reading-1
	2	Concept Learning: introduction, a concept learning task, concept learning as search; general to specific ordering of hypotheses; finding a maximally specific hypothesis	Ch#2	
2	3	Version Spaces: basics, If-then-eliminate algorithm		Reading-2
	4	Candidate elimination algorithm		
3	5	Inductive bias, effect on hypothesis space		Quiz-1
	6	Supervised Learning: introduction; Decision Tree learning	Ch#3	
4	7	Selecting best attribute, extracting rules from learned decision trees		
	8	Issues in decision tree learning		Assignment-1
5	9	Bayesian approach to classification: MAP, Naïve Bayes assumption	Ch#6	
	10	Naïve Bayes classifier		
6	11	Artificial Neural Networks: Perceptrons	Ch#4	Quiz-2
	12	Multilayer Perceptron networks and the backpropagation algorithm		
7	13	Issues in MLP training; performance enhancements for backpropagation algorithm		Reading-3
	14	Kernel methods: Support Vector Machines	Handouts	Assignment-2
8	15	SVM training; kernel functions, Issues in multiclass problems	Handouts	
	16	Midterm review		
		Midterm Exam		
9	17	Linear models for regression and classification	Handouts	Assignment-3
	18	Linear regression		
10	19	Logistic regression	Handouts	Reading-4
	20	Logistic regression (contd.)		
11	21	Unsupervised learning: basics		Quiz-3
	22	Clustering methods: partition based; K-means algorithm	Handouts	

Week	**Lecture**	**Topic**	**Source Book (Ch#)**	**Recommendation for Learning Activities**
12	23	Hierarchical clustering; agglomerative vs divisive methods		Assignment-4
	24	Hierarchical clustering (contd.)		
13	25	Self-organizing maps (SOMs)	Handouts	Reading-5
	26	Semi supervised learning; basics; applications	Handouts	
14	27	Expectation maximization algorithm		Reading-6
	28	Reinforcement Learning: introduction	Ch#13	Quiz-4
15	29	HMMs; Monte Carlo inference		
	30	Exploration vs exploitation trade-off; Q-Learning		Assignment-5
16	31	Ensemble Learning: Committees; Boosting; Bagging	Handouts	
	32	Final term review		
Final Exam				

<table>
<tr><td>Program</td><td colspan="3">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="3">ED-321</td></tr>
<tr><td>Course Title</td><td colspan="3">Deep Learning</td></tr>
<tr><td rowspan="2">Credit Hours</td><td colspan="2">Theory</td><td>Lab</td></tr>
<tr><td colspan="2">3</td><td>Nil</td></tr>
<tr><td>Lecture Duration</td><td colspan="3">90 minutes (1.5 Hours), 2 lectures per week</td></tr>
<tr><td>Semester</td><td colspan="3">8</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td colspan="2">Courses</td><td>Knowledge</td></tr>
<tr><td colspan="2">Nil</td><td>Nil</td></tr>
<tr><td>Follow Up Courses</td><td colspan="3">NilL</td></tr>
<tr><td colspan="4">Course Learning Outcomes (CLOs)</td></tr>
<tr><td>CLO No</td><td colspan="2">Course Learning Outcome</td><td>Bloom Taxonomy</td></tr>
<tr><td>CLO-1</td><td colspan="2">Apply deep learning algorithms to real-world problems</td><td>C3 (Apply)</td></tr>
<tr><td>CLO-2</td><td colspan="2">Analyze results from deep learning to select appropriate solutions</td><td>C4 (Analyze)</td></tr>
<tr><td>CLO-3</td><td colspan="2">Code the novel neural network architectures from scratch and evaluating the performance on application specific standard benchmarks</td><td>C3 (Apply)</td></tr>
<tr><td>Aims and Objectives</td><td colspan="3">1. Understand the Deep Learning concepts
2. Apply deep learning algorithms to real-world problems
3. Analyze results from deep learning to select appropriate solutions
4. Code the novel neural network architectures from scratch and evaluating the performance on application specific standard benchmarks</td></tr>
<tr><td>Learning Outcomes</td><td colspan="3">• Understand the Deep Learning concepts</td></tr>
</table>

	• Apply deep learning algorithms to real-world problems • Analyze results from deep learning to select appropriate solutions • Code the novel neural network architectures from scratch and evaluating the performance on application specific standard benchmarks
Syllabus	Basics of deep learning, learning networks, Shallow vs. Deep learning etc.; Machine learning theory – training and test sets, evaluation, etc. Theory of Generalization; Multi-layer perceptrons, error back-propagation; Deep convolutional networks, Computational complexity of feed forward and deep convolutional neural networks; Unsupervised deep learning including auto-encoders; Deep belief networks; Restricted Boltzman Machines; Deep Recurrent Neural Networks (BPTT, LSTM, etc.); GPU programming for deep learning CuDNN; Generative adversarial networks (GANs); Sparse coding and auto-encoders; Data augmentation, elastic distortions, data normalization; Mitigating overfitting with dropout, batch normalization, dropconnect; Novel architectures, ResNet, GoogleNet, etc
Contents	1. Basics of deep learning, 2. Learning networks, 3. Shallow vs. Deep learning etc. 4. Machine learning theory 4.1. training and test sets, 4.2. evaluation, etc. 5. Theory of Generalization 6. Multi-layer perceptrons, 7. Error back-propagation 8. Deep convolutional networks, 8.1. Computational complexity of feed forward and deep convolutional neural networks

<table>
<tr><td></td><td>9. Unsupervised deep learning including auto-encoders
10. Deep belief networks
11. Restricted Boltzman Machines
12. Deep Recurrent Neural Networks (BPTT, LSTM, etc.)
13. GPU programming for deep learning CuDNN
14. Generative adversarial networks (GANs)
15. Sparse coding and auto-encoders
16. Data augmentation,
17. Elastic distortions
18. Data normalization
19. Mitigating overfitting with dropout
20. Batch normalization,
21. Dropconnect
22. Novel architectures,
23. ResNet,
24. GoogleNet, etc</td></tr>
<tr><td>Teaching-learning Strategies</td><td>• Interactive class session
• Hands on practices in class
• Brainstorming and Group discussion sessions</td></tr>
<tr><td>Assignments</td><td>• Paper based written assignments 2
• Programming Assignments 6</td></tr>
<tr><td>Assessment and Examinations</td><td><table><tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr><tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr></table></td></tr>
</table>

	2	Midterm Assessment	35%	It takes place at the mid-point of the semester.
	3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.
Textbooks	• Deep Learning by Ian Goodfellow, Yoshua Bengio, Aaron Courville (http://www.deeplearningbook.org/) • Deep learning with python by Francoise Chollet, ISBN-10: 9781617294433, 2017			
Reference Material/Suggested Readings	• Handout provided by the teacher. • PowerPoint Presentations • Various books Chapters / Notes • Internet resources			
Notes	• Academic integrity is expected of all students. Plagiarism or cheating in any assessment will result in at least an F grade in the course, and possibly more severe penalties. • You bear all the responsibility for protecting your assignments from plagiarism. If anyone else submits your assignment or uses your code in his/her assignment, you will be considered equally responsible. • The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester.			

	• There is no makeup for a missed sessional grading instruments like quizzes, assignments, and homework's.

Program	BS Data Science	
Course Code	ED-345	
Course Title	Artificial Neural Network	
Credit Hours	**Theory**	**Lab**
	2	1
Lecture Duration	60 minutes (1 Hours), 2 lectures per week, 3 hours lab session per week	
Semester	7	
Pre-requisites	**Courses**	**Knowledge**
	Machine Learning	Nil
Follow Up Courses	Deep Learning	

Course Learning Outcomes (CLOs)		
CLO No	**Course Learning Outcome**	**Bloom Taxonomy**
CLO-1	Understand the fundamentals of neural networks in AI.	C2 (Understand)
CLO-2	Explain how simple ANNs can be designed.	C2 (Understand)
CLO-3	Apply ANN for complex problems.	C3 (Apply)

Aims and Objectives	1. Understand the fundamentals of neural networks in AI 2. Explain how simple ANNs can be designed. 3. Apply ANN for classification Problems 4. Differentiate between different Networks and their learning laws
Learning Outcomes	• Understand the fundamentals of neural networks in AI • Explain how simple ANNs can be designed. • Apply ANN for classification Problems • Differentiate between different Networks and their learning laws

Syllabus	Introduction and history of neural networks, Basic architecture of neural networks, Perceptron and Adaline (Minimum Error Learning) for classification, Gradient descent (Delta) rule, Hebbian, Neo-Hebbian and Differential Hebbian Learning, Drive Reinforcement Theory, Kohonen Self Organizing Maps, Associative memory, Bi-directional associative memory (BAM), Energy surfaces, The Boltzmann machines, Backpropagation Networks, Feedforward Networks; Introduction to Deep learning and its architecture
Contents	1. Introduction and history of neural networks, 2. Basic architecture of neural networks, 3. Perceptron and Adaline (Minimum Error Learning) for classification, 4. Gradient descent (Delta) rule, 5. Hebbian, Neo-Hebbian and Differential Hebbian Learning, 6. Drive Reinforcement Theory, 7. Kohonen Self Organizing Maps, 8. Associative memory, 9. Bi-directional associative memory (BAM), 10. Energy surfaces, 11. The Boltzmann machines, 12. Backpropagation Networks, 13. Feedforward Networks; 14. Introduction to Deep learning and its architecture
Teaching-learning Strategies	• Interactive class session • Hands on practices in class • Brainstorming and Group discussion sessions
Assignments	• Paper based written assignments 4 • Programming Assignments 6

	Sr. #	Elements	Weightage	Details

Assessment and Examinations	1	Formative Assessment	25%	It is continuous assessment. It includes classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.
	2	Midterm Assessment	35%	It takes place at the mid-point of the semester.
	3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.
Textbooks	• Neural Network Design, 2nd Edition, Martin T. Hagan, Howard, B. Demuth, Mark Hudson Beale and Orlando De Jesus, Publisher: Martin Hagan; 2 edition (September 1, 2014), ISBN-10: 0971732116 • An Introduction to Neural Networks, James A Anderson, Publisher: A Bradford Book (March 16, 1995), ISBN-10: 0262011441 • Fundamentals of Artificial Neural Networks, Mohammad Hassoun, Publisher: A Bradford Book (January 1, 2003), ISBN-10: 0262514672			
Reference Material/Suggested Readings	• Handout provided by the teacher. • PowerPoint Presentations • Various books Chapters / Notes • Internet resources			

Notes	• Academic integrity is expected of all students. Plagiarism or cheating in any assessment will result in at least an F grade in the course, and possibly more severe penalties. • You bear all the responsibility for protecting your assignments from plagiarism. If anyone else submits your assignment or uses your code in his/her assignment, you will be considered equally responsible. • The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester. • There is no makeup for a missed sessional grading instruments like quizzes, assignments, and homework's.

<table>
<tr><td>Program</td><td colspan="2">BS data science</td></tr>
<tr><td>Course Code</td><td colspan="2">DS-3XX</td></tr>
<tr><td>Course Title</td><td colspan="2">Business Process Management</td></tr>
<tr><td rowspan="2">Credit Hours</td><th>Theory</th><th>Lab</th></tr>
<tr><td>3</td><td>0</td></tr>
<tr><td>Lecture Duration</td><td colspan="2">90 minutes (1.5 Hours), 2 lectures per week</td></tr>
<tr><td>Semester</td><td colspan="2">7</td></tr>
<tr><td rowspan="2">Pre-requisites</td><th>Courses</th><th>Knowledge</th></tr>
<tr><td></td><td>Nil</td></tr>
<tr><td>Follow Up Courses</td><td colspan="2"></td></tr>
<tr><td>Aims and Objectives</td><td colspan="2">Upon completion of this course, the student will be able to:

1. Use business process modeling tools
2. Design and model business processes
3. Perform process mining</td></tr>
<tr><td>Learning Outcomes</td><td colspan="2">At the end of the course, you should be able to:
• Use Design and model business processes Explain the model classification at different levels.
•</td></tr>
<tr><td>Contents</td><td colspan="2">1. Introduction and background
1.1. Process identification
1.2. Overview of modeling
1.3. BPM lifecycle
1.4. Definition of process architecture
1.5. Process selection
2. Essentials of process modeling</td></tr>
</table>

	2.1. Branching and merging 2.2. Business objects 2.3. Resources 2.4. Process decomposition 2.5. Process model reuse 3. Advanced process modeling 3.1. Rework and repetition 3.2. Handling events 3.3. Handling exceptions 3.4. Business rules 4. Process discovery 4.1. Process discovery methods 4.2. Process modeling method 4.3. Process model quality assurance 5. Qualitative process analysis 5.1. Value added analysis 5.2. Waste analysis 5.3. Stakeholder analysis and issue documentation 5.4. Root cause analysis 6. Process redesign 6.1. Transactional methods 6.2. Transformational methods 7. Process monitoring 7.1. Context of monitoring 7.2. Process performance dashboards 7.3. Introduction to mining
Teaching-learning Strategies	• Interactive class session • Hands on practices in class • Brainstorming and Group discussion sessions

<table>
<tr><td></td><td></td></tr>
<tr><td>Assignments</td><td>Paper based written assignments 3
Project 2
Quiz 4</td></tr>
<tr><td rowspan="4">Assessment and Examinations</td><td>
<table>
<tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper.</td></tr>
</table>
</td></tr>
<tr><td>Textbooks & Reference material</td><td>Marlon Dumas, Marcello La Rosa, Jan Mendling, Hajo Reijers. Fundamentals of Business process Management. Springer, 2nd Edition, 2017
Mathias Weske. Business Process Management: Concepts, Languages, Architectures. Springer, 2015.</td></tr>
<tr><td>Notes</td><td>The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester.</td></tr>
</table>

<table>
<tr><td>Program</td><td colspan="2">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="2">DS-4XX</td></tr>
<tr><td>Course Title</td><td colspan="2">Cloud Computing</td></tr>
<tr><td rowspan="2">Credit Hours</td><td>Theory</td><td>Lab</td></tr>
<tr><td>3</td><td>0</td></tr>
<tr><td>Lecture Duration</td><td colspan="2">90 minutes (1.5 Hours) 2 lectures per week</td></tr>
<tr><td>Semester</td><td colspan="2">7</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td>Courses</td><td>Knowledge</td></tr>
<tr><td>Nil</td><td>Nil</td></tr>
<tr><td>Follow Up Courses</td><td colspan="2"></td></tr>
<tr><td>Objectives</td><td colspan="2">1. Cloud computing is attractive for individuals users as well as for the businesses. It provides on-demand computation and storage resources, which is attractive for a large number of users, mainly due to pay-per-usage charging model.
2. This course will introduce topics, tools, and technologies of
3. cloud computing to the students. The course would also expose students to the essential tools and technologies used in Cloud Computing.</td></tr>
<tr><td>Learning Outcomes</td><td colspan="2">• Understand of cloud computing
• Hands-on experience using cloud resources
• Have knowledge of cloud technologies
• Understand distributed systems concepts</td></tr>
<tr><td>Syllabus</td><td colspan="2">The syllabus of the course covers</td></tr>
</table>

	Introduction to Cloud Computing, AWS Services, EC2 handson,Introduction to Scraping, Selenium, Python Scrapy, Introduction to Virtualization, Introduction to Big Data, Introduction to NoSQL technologies (MongoDB and DynamoDB), MapReduce/Hadoop, Apache Spark, Serverless Computing, Introduction to Blockchain, Autoscaling Cloud Applications, Consistency in Distributed Systems, Fault Tolerance in Distributed Systems, Fault Tolerance in Distributed Systems,
Contents	1. Introduction to Cloud Computing 1.1 Definition and History of Cloud Computing 1.2 Service Models 1.3 Deployment Models 2. AWS Services, EC2 hands-on 2.1 S3 2.2 EC2 2.3 RDS 2.4 Dynamo DB 3. Introduction to Scraping, Selenium, Python Scrapy 3.1 Introduction to Scraping 3.2 Using Python for Scrapping 4. Virtualization 4.1 Introduction to Virtualization 4.2 Host Viruatlization 4.3 Paravirtualization 4.4 Hardware Virtualization 4.5 Introduction to Containers

	5. Big Data 5.1 Introduction to Big Data and its Characteristics 5.2 Big Data Processing Models 5.3 Big Data Analytics Types and Uses 6. NoSQL technologies 6.1 MongoDB 6.2 DynamoDB 6.3 RADIS 7. Batch Processing using MapReduce/Hadoop 8. Stream Processing using Apache Spark 9. Serverless Computing 10. Introduction to Blockchain 11. Autoscaling Cloud Applications 12. Consistency in Distributed Systems, 13. Fault Tolerance in Distributed Systems, 14. Fault Tolerance in Distributed Systems
Teaching-learning Strategies	• Interactive class session • Hands on assignments and tutorials • Group project
Assignments	• Practical Assignments 5

	Sr. #	Elements	Weightage	Details
Assessment and Examinations	1	Formative Assessment	25%	It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude

				and behavior, hands-on-activities, short tests, quizzes etc.
	2	Midterm Assessment	35%	It takes place at the mid-point of the semester.
	3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.
Textbooks	• Jothy Rosenberg and Arthur Mateos; The Cloud at Your Service; Manning Publications. ISBN: 1935182528			
Reference Material/Suggested Readings	• Paul Zikopoulos and Chris Eaton; Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data; McGraw-Hill. ISBN: 0071790535 • Kyle Banker, Peter Bakkum, Shaun Verch, Douglas Garrett, and Tim Hawkins; MongoDB in Action, Second Edition. • Clinton W. Brownley; Foundations for Analytics with Python from Non- Programmer to Hacker.			

Notes	• Academic integrity is expected of all students. Plagiarism or cheating in any assessment will result in at least an F grade in the course, and possibly more severe penalties. • You bear all the responsibility for protecting your assignments from plagiarism. If anyone else submits your assignment or uses your code in his/her assignment, you will be considered equally responsible. • The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester. • There is no makeup for a missed sessional grading instruments like quizzes, assignments, and homework's.

Detailed Lecture wise plan

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
1	1	Introduction To Cloud Computing		
	2	Introduction To Cloud Computing (Cont.)		Assignment
2	3	Introduction To Big Data		
	4	Introduction To Big Data (Cont.)		Quiz
3	5	Virtualization		
	6	Containerization/Dockers		
4	7	Fundamental concepts of Distributed Systems		Assignment
	8	Fundamental concepts of Distributed Systems (Cont.)		
5	9	Fundamental concepts of Distributed Systems (Cont.)		Quiz
	10	Fundamental concepts of Distributed Systems (Cont.)		
6	11	MapReduce		Quiz
	12	MapReduce Programming and Case Studies		
7	13	Scalable Web Application in The Cloud		
	14	Scalable Web Application in The Cloud (Cont.)		Assignment

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
8	15	Big Data Processing Methods and Techniques		
	16	Apache Kafka: Distributed Messaging Systems		
Midterm Exams				
9	17	Data Analytics Using Python		Quiz
	18	Data Analytics Using Python (Cont.)		
10	19	Introduction to Apache Spark		
	20	Using Apache Spark Solving Real Problems: Case Studie		
11	21	Introduction to NoSQL Databases		
	22	MongoDB and Apache Cassandra		Assignment
12	23	RADIS		
	24	Big Data Transformation Methods		
13	25	Big Data Transformation Methods (Cont.)		Assignment
	26	Predictive Analytics: Applied Machine Learning		
14	27	Practical Considerations in Cloud Computing		
	28	Future of the Cloud Computing and Big Data		Quiz
15	29	Introduction to Blockchain		

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
	30	Recent advancements in Cloud Computing		
16	31	Research Topics in Cloud Computing		Research Papers
	32	Research Topics in Cloud Computing		Research Papers
Final Exam				

<table>
<tr><td>Program</td><td colspan="2">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="2">ED-441</td></tr>
<tr><td>Course Title</td><td colspan="2">Visual Programming</td></tr>
<tr><td rowspan="2">Credit Hours</td><td colspan="2">Theory</td></tr>
<tr><td colspan="2">3</td></tr>
<tr><td>Lecture Duration</td><td colspan="2">90 minutes (1.5 Hours), 2 lectures per week</td></tr>
<tr><td>Semester</td><td colspan="2">5</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td>Courses</td><td>Knowledge</td></tr>
<tr><td>Object Oriented Programming</td><td>• Practical experience and proficiency with C/C++ or Java.
• Good concepts and knowledge of OOP and Design Principles.
• Basic knowledge about server and client side processing architecture.</td></tr>
<tr><td>Follow Up Courses</td><td colspan="2">NA</td></tr>
<tr><td>Objectives</td><td colspan="2">1. Understanding of Visual C# Programming
2. Understanding of Programming in C# .NET
3. Understanding Microsoft .NET architecture.
4. Understanding and developing Software Development using Microsoft Platform/Technologies.</td></tr>
</table>

	5. To have solid working experience C# programming language 6. To be efficient in developing desktop and web applications using Microsoft .NET framework and class library
Learning Outcomes	• Student should be able to design desktop and web applications. • Student should be able to understand the visual programming. • Student can develop real time software. • Student should have practical knowledge of developing applications using N-Tier Architecture, Service Oriented Architecture, Loosely Coupled Systems, MVC Architecture and Single Page Applications.
Syllabus	Introduction to Course, Overview of Visual Programming C#, Microsoft technology history, Intro to .net and its architecture, Concept of MSIL, CLR, CLS, CTS, HelloWorld Program, compilation through command line using csc compiler,.NET Managed and Unmanaged Code, Assembly file Concept, Type of Assembly (exe, dll).Global Assembly Cache, Assembly manifest , input from command line, Input from Console. Creation of Assembly file(dll, exe) using csc compiler., Introduction to C#, Data Types, value Types, Reference Types, Control Structures, Loops, foreach loop, C# Class structure and Access specifiers(Public, private), Object creation, Concept of Namespaces, ref and var keyword, Boxing and Un-Boxing, Using out and dynamic keywords, params modifier, Object Initializers ,Optional Arguments, Named Arguments, Namespace Aliasing-Tier Architecture Concept, use of Data access, business objects, business logic and presentation layer in N-Tier. Implementation of N-Tier, Introduction to ADO.Net, Connected .Net Data Providers(Connection, Command, DataReader), Generics, Collections (List, Dictionary),Sql Injection, parameterized queries. Delegate, Introduction to WPF, XAML Basics, Multicast delegates, Anonymous Methods, Lambda Expression, WPF layout Concept, StackPanel, Element Binding, Events, Dock and Grid Layout, Dependency and attach Properties, Introduction to

	Disconnected Model, Usage of Data Set, Data Adapter and Command Builder in disconnected Model. Data Binding, Simple and Complex Data Binding, , List and Grids, Microsoft Reports, HTML, Introduction to java script, data types, variables, functions, Debugging js using Firebug, DOM Tree., Introduction to the Browser's Object (BOM), Events, Event Flow, Event Capturing vs Event bubbling, Query selector API,Introduction to Jquery, Selecting and Filtering, Event, Manipulating Contents and Attributes, Jquery Effects, Manipulating CSS, AJAX, Introducing LINQ, LINQ to Objects, Query Syntax,LINQ to Sql, Projection, Filtering and Join In Linq Queries, Method Syntax, Extension Methods, Lambda Expression, Introduction to ASP.NET MVC, MVC Application Structure, Controllers overview, Action Methods, parameterized action methods, Introduction to Razor Syntax, Code Expressions, , Code Blocks, Implicit Vs Explicit Code Expression, Views, ViewData and ViewBag, Strongly Typed Views,View Models, Layouts, ViewStart, partial Views, Model, Model Binding, Introduction to ADO.NET Entity Framework, The Entity Data Model, CSDL: The Conceptual Schema, SSDL: The Store Schema, MSL: The Mappings,Eager vs Lazy Loading, POCO Classes, DBContext API, Querying Entity Data Models, LINQ to Entities, Projection, Navigation, Joins in queries, Modifying Entities and Saving Changes, Forms, Get Vs Post, Html Helpers, Form, Input Helpers, Strongly Typed Helpers, Templated Helpers, Helpers and Model State, Data Annotations, Client + Server Side Validation, Validation and Model Binding, Validation and Model State, Building Loosely Coupled Components, Introduction to dependency Injection, Constructor Injection, D.I using NInject., Dependency Inject in MVC, Introduction to Repository Pattern. Introduction to Service Oriented Architecture, WSDL, Service Contract, Data Contract, XML, Example: Car Rental Service, WCF Bindings, ABC of WCF, Restful Services, Consuming rest services (CRUD operations) using Jquery AJAX and JSON., Introduction to Web API, Example of Web API using CRUD

	Example, MVC Memebership, Authorization and Security, Asp.Net Identity, MVC Routing, Deployment of Web Application, Profiling of application, Onion Architecture, Introduction, Advantages of Onion Architecture, Onion Architecture Layers, Single-page application, Introduction to Blazor Tutorial - Build your first Blazor application
Contents	1. Introduction to Visual Programming 2. Introduction to C# 3. Microsoft technology history, Intro to .net and its architecture, 3.1. Concept of MSIL, CLR, CLS, CTS, HelloWorld **4.** .NET Managed and Unmanaged Code **5.** Intro to C# 5.1. Classes data types, Access specifiers 5.2. Boxing unboxing, namespace **6.** N-tier Architecture **7.** ADO.Net **8.** SQL Injection 9. Delegates **10.** WPF, Desktop application development 11. HTML, Javscript, CSS 12. Jquery 13. LINQ to SQL 14. ASP.NET MVC 15. Razor syntax 16. Web APIs 17. Onion Architecture 18. Single page applications
Teaching-learning Strategies	• Interactive class session • Hands on practices in class

<table>
<tr><td></td><td>• Brainstorming and Group discussion sessions
• Coding in LABS</td></tr>
<tr><td>Assignments</td><td>Coding Assignments 6</td></tr>
<tr><td>Assessment and Examinations</td><td>
<table>
<tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.</td></tr>
</table>
</td></tr>
<tr><td>Textbooks</td><td>A. Schildt, H. (2010). C# 4.0: The complete reference. Tata McGraw-Hill Education.
B. Freeman, A., & Sanderson, S. (2013). Pro Asp. net Mvc 4 (Vol. 832). Apress.
C. Galloway, J., Haack, P., Wilson, B., & Allen, K. S. (2012). Professional ASP. NET MVC 4. John Wiley & Sons.
D. Lerman, J. (2010). Programming Entity Framework: Building Data Centric Apps with the ADO. NET Entity Framework. " O'Reilly Media, Inc.".
E. LINQ in Action by MANNING
F. Wilton, P. (2004). Beginning JavaScript. John Wiley & Sons.</td></tr>
</table>

	G. York, R. (2011). Beginning JavaScript and CSS development with jQuery. John Wiley & Sons. H. Cibraro, P., Claeys, K., Cozzolino, F., & Grabner, J. (2010). Professional WCF 4: Windows communication foundation with. NET 4. John Wiley & Sons. I. MacDonald, M. (2012). Pro WPF 4.5 in VB. Apress. J. Team, W. A. (2001). Professional ADO .NET. K. Galloway, J., Haack, P., Wilson, B., & Allen, K. S. (2012). Professional ASP. NET MVC 4. John Wiley & Sons.
Reference Material/Suggested Readings	• **R1.** Handouts • **R2.** Deitel, P., & Deitel, H. (2017). Visual C# how to program. Pearson. • **R3.** Microsoft Visual C# 2013 Step by Step (Step by Step Developer), Sharp, J., 1st Edition (2013), Microsoft Press. • **R4.** http://www.asp.net/web-api • **R5.** http://www.msdn.com

Detailed Lecture wise plan

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
1	1	Introduction to Course, Overview of Visual Programming, Microsoft Visual C#, Microsoft technology history, Intro to .net and its architecture, Concept of MSIL, CLR, CLS, CTS, HelloWorld Program, compilation through command line using csc compiler	R1: Reading Material A-Chap 1 pg(3-8) A-Chap 2 pg(11-23) R2	
	2	.NET Managed and Unmanaged Code, Assembly file Concept, Type of Assembly (exe, dll).Global Assembly Cache, Assembly manifest , input from command line, Input from Console. Creation of Assembly file(dll, exe) using csc compiler.	R1-(Reading Material) A-Chap 1 pg(8-9) A-Chapter 2 pg (13- pg-28)	
2	3	Introduction to C#, Data Types, value Types, Reference Types, Control Structures, Loops, foreach loop, C# Class structure and Access specifiers(Public, private), Object creation, Concept of Namespaces, ref and var keyword	A-Chap 3, Chap 5, Chap 6 pg (111-131), Chap 8 pg (167-174),Chap 16 pg (449-463)	Quiz#1
	4	Boxing and Un-Boxing, Using out and dynamic keywords, params modifier, Object Initializers ,Optional Arguments, Named Arguments, Namespace Aliasing	A-Chap 11 pg 315 A-Chap 6 Pg 132-135,Chap 3 pg(51) ,Chap– 8 pg (167-210)	
3	5	N-Tier Architecture Concept, use of Data access, business objects, business logic and presentation layer in N-Tier. Implementation of N-Tier	R1	Quiz#2 Assign-1
	6	Introduction to ADO.Net, Connected .Net Data Providers(Connection, Command, DataReader), Generics, Collections (List, Dictionary)	J- pre chapter content pg(1-6),chap 1(7-40), Chap 2 pg(45-60)	

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
4	7	Sql Injection, parameterized queries. Delegate, Introduction to WPF, XAML Basics	R1 K-chap 7 pg(137-187) I-chap 3 pg(43-73)	
	8	Multicast delegates, Anonymous Methods, Lambda Expression, WPF layout Concept, StackPanel, Element Binding	A-Chap15 pg(411-445)	Quiz#3
5	9	Events, Dock and Grid Layout, Dependency and attach Properties, Introduction to Disconnected Model	A-Chap15 pg(411-445) J- Chap 9 pg(317-365)	
	10	Usage of Data Set, Data Adapter and Command Builder in disconnected Model. Data Binding, Simple and Complex Data Binding, , List and Grids, Microsoft Reports	R1,W2, J-Chap 6 pg(207-235) J-Chap 9 pg(317-365)	Quiz#4 Assign-2
6	11	Html, Introduction to java script,data types, variables, functions, Debugging js using Firebug, DOM Tree.	J-chap 13 pg(455) F-Chap1	
	12	Introduction to the Browser's Object (BOM), Events, Event Flow, Event Capturing vs Event bubbling, Query selector API	K-chap 8 (pg 189) F-Chap12	Quiz#5
7	13	Introduction to Jquery, Selecting and Filtering, Event, Manipulating Contents and Attributes	K-chap 8(pg 189) G-Chap1-4	
	14	Jquery Effects, Manipulating CSS, AJAX, Introducing LINQ, LINQ to Objects, Query Syntax	K-chap 8(pg 189) A-chap 19(pg 565) G-Chap7-8	
8	15	LINQ to Sql, Projection, Filtering and Join In Linq Queries, Method Syntax	A-chap 19(pg 565) E-Chap1, E-Chap3	Quiz#6
	16	Extension Methods, Lambda Expression, Revision	A-chap 19(pg 597) I-chap 2	Assign-3

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
9	17	Introduction to ASP.NET MVC, MVC Application Structure, Controllers overview, Action Methods, parameterized action methods	I-chap 1,2(pg 3,15) K-chap 1(pg 1)	Quiz#7
	18	Introduction to Razor Syntax, Code Expressions, , Code Blocks, Implicit Vs Explicit Code Expression, Views, ViewData and ViewBag, Strongly Typed Views,View Models	I-chap 5 (pg 101) I-chap 8(pg 201) I-chap 18(pg 485) K-chap3 (pg 47)	
10	19	Layouts, ViewStart, partial Views, Model, Model Binding, Introduction to ADO.NET Entity Framework, The Entity Data Model, CSDL: The Conceptual Schema, SSDL: The Store Schema, MSL: The Mappings	I-chap 5 (pg 101) K-chap3 (pg 47) I-chap 2 (pg 15)	
	20	Eager vs Lazy Loading, POCO Classes, DBContext API, Querying Entity Data Models, LINQ to Entities, Projection, Navigation, Joins in queries, Modifying Entities and Saving Changes	I-Chap 8 (pg 201) K-chap 4 (pg 71)	Assign-4
11	21	Forms, Get Vs Post, Html Helpers, Form, Input Helpers, Strongly Typed Helpers, Templated Helpers, Helpers and Model State	K-chap 5 (pg 95)	Quiz#8
	22	Data Annotations, Client + Server Side Validation, Validation and Model Binding, Validation and Model State	K-chap 6 (pg 119)	
12	23	Building Loosely Coupled Components, Introduction to dependency Injection, Constructor Injection, D.I using NInject.	K-chap 16(pg 423) I-chap 3(pg 47) K-chap 12(pg 297)	
	24	Dependency Inject in MVC, Introduction to Repository Pattern.	I-chap 3(pg 47) K-chap 12(pg 297)	
13	25	Introduction to Service Oriented Architecture, WSDL, Service Contract, Data Contract, XML, Example: Car Rental Service	K-chap 11 (pg 279) J-chap 13(455) I chap 1(pg 6)	Quiz#9
	26	WCF Bindings, ABC of WCF, Restful Services, Consuming rest services (CRUD operations) using Jquery AJAX and JSON.	H-chap 4(pg 103) H-chap 1(pg 1) H-chap 2(pg 33) J-chap 15(pg 551)	Assign-5

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
14	27	Introduction to Web API, Example of Web API using CRUD Example	K-chap 11(279) H-chap 4(pg 103) J-chap 15(pg 551) I-chap 12(pg 303) I-chap 09(pg 229)	Quiz#10
	28	MVC Memebership, Authorization and Security, Asp.Net Identity	K-chap 7(pg 137) I-chap 11(pg 283)	
15	29	MVC Routing, Deployment of Web Application, Profiling of application	K-chap 11(279) K-chap 7(pg 137) K-chap 16 (pg 423) I-chap 24 (pg 657) I-chap 26 (pg 699)	
	30	Onion Architecture, Introduction, Advantages of Onion Architecture, Onion Architecture Layers	R1	
16	31	Single-page application, Introduction to Blazor Tutorial - Build your first Blazor application	R1	
	32	Blazor (con.)	R1	

<table>
<tr><td>Program</td><td colspan="2">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="2">ED-323</td></tr>
<tr><td>Course Title</td><td colspan="2">Web Technologies</td></tr>
<tr><td rowspan="2">Credit Hours</td><td>Theory</td><td>Lab</td></tr>
<tr><td>2</td><td>1</td></tr>
<tr><td>Lecture Duration</td><td colspan="2">60 minutes (1 Hours), 2 lectures per week, 3 hours lab session per week</td></tr>
<tr><td>Semester</td><td colspan="2">5</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td>Courses</td><td>Knowledge</td></tr>
<tr><td>Nil</td><td>Nil</td></tr>
<tr><td>Follow Up Courses</td><td colspan="2">Nil</td></tr>
<tr><td>Aims and Objectives</td><td colspan="2">1. The course of Web Technologies is particularly designed to give students a broad understanding of a structured methodology which is utilized in software engineering to Web development projects.
2. The course addresses the concepts, methods, technologies, and techniques of developing Web sites that collect, organize and expose information resources.
3. This course introduces students to the discipline of Web Technologies including the methods and techniques used in web-based system development.
4. In contrast to traditional software engineering, Web Technologies methods and techniques must incorporate unique aspects of the problem domain such as: document-oriented delivery, fine-grained lifecycles, user-centric development,</td></tr>
</table>

	client-server legacy system integration and diverse end user skill levels. 5. This course draws upon previous programming and computing experience to develop practical web development and maintenance skills. 6. This course is intended for students with knowledge of both Internet communication concepts and an introductory programming knowledge.
Learning Outcomes	• On successful completion of the course students will be able to: 1. Develop a web application using server-side programming languages and components. 2. Apply the Web Technologies methodologies for Web application development 3. Develop a component-based web solution and use UML diagrams to describe such a solution. 4. Identify and discuss the security risk of a Web application.
Syllabus	• Introduction • Working on Presentation Tier • Display web contents (HTML) • Styling web content (CSS) • Controlling Behavior of Content (JavaScript) • Design pattern in JavaScript • Composite pattern (jQuery) • MVC (AngularJS) • Component based (React) • Working on Logical tier/ Application tier • Dynamic Server pages (JSP) • Java Servlets • CURD operations in Java

<table>
<tr><td></td><td>MVC Design pattern using Spring Framework
ORM using Hibernate</td></tr>
<tr><td>Contents</td><td>Design methodologies to support web-based software systems
Deployment and maintenance models for web-based software systems
Server-side programming and web application frameworks
System security for web-based software systems
Techniques to support mobile devices</td></tr>
<tr><td>Teaching-learning Strategies</td><td>Interactive class session
Hands on practices in class
Brainstorming and Group discussion sessions
Coding in LABS</td></tr>
<tr><td>Assignments</td><td>Paper based written assignments 8</td></tr>
<tr><td>Assessment and Examinations</td><td>
<table>
<tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, short tests, quizzes etc.</td></tr>
<tr><td>2</td><td>Midterm Assessment</td><td>35%</td><td>It takes place at the mid-point of the semester.</td></tr>
<tr><td>3</td><td>Final Assessment</td><td>40%</td><td>It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based</td></tr>
</table>
</td></tr>
</table>

				on term paper, research proposal development, field work and report writing etc.
Textbooks				
Reference Material/Suggested Readings	• Handout provided by the teacher. • PowerPoint Presentations • Various books Chapters / Notes • Internet resources			
Notes	• Academic integrity is expected of all students. Plagiarism or cheating in any assessment will result in at least an F grade in the course, and possibly more severe penalties. • You bear all the responsibility for protecting your assignments from plagiarism. If anyone else submits your assignment or uses your code in his/her assignment, you will be considered equally responsible. • The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester. • There is no makeup for a missed sessional grading instruments like quizzes, assignments, and homework's.			

Detailed Lecture wise plan

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
1	1	Introduction		
	2	Introduction		
2	3	Working on Presentation Tier		
	4	Display web contents (HTML)		
3	5	Display web contents (HTML)		
	6	Display web contents (HTML)		
4	7	Display web contents (HTML)		
	8	Styling web content (CSS)		
5	9	Styling web content (CSS)		
	10	Styling web content (CSS)		
6	11	Styling web content (CSS)		
	12	Controlling Behavior of Content (JavaScript)		
7	13	Controlling Behavior of Content (JavaScript)		
	14	Controlling Behavior of Content (JavaScript)		
8	15	Controlling Behavior of Content (JavaScript)		
	16	Design pattern in JavaScript		
Midterm Exams				
9	17	Composite pattern (jQuery)		
	18	Composite pattern (jQuery)		
10	19	MVC (AngularJS)		
	20	MVC (AngularJS)		
11	21	Component based (React)		
	22	Component based (React)		
12	23	Working on Logical tier/ Application tier		
	24	Dynamic Server pages (JSP)		
13	25	Java Servlets		
	26	Java Servlets		
14	27	CURD operations in Java		
	28	CURD operations in Java		
15	29	MVC Design pattern using Spring Framework		
	30	MVC Design pattern using Spring Framework		
16	31	ORM using Hibernate		
	32	ORM using Hibernate		
Final Exam				

Program	BS Data Science	
Course Code	ED-442	
Course Title	Systems Programming	
Credit Hours	**Theory**	**Lab**
	3	
Lecture Duration	90 minutes (1.5 Hours), 2 lectures per week	
Semester		
Pre-requisites	**Courses**	**Knowledge**
	Operating Systems	
Follow Up Courses	7	
Aims and Objectives	1. To understand how different services of an operating system works, and write different OS services of your own. 2. Understand the working of OS Kernel, write its patches and compile the Kernel source and install it on bare hardware	
Learning Outcomes	• Students will be able to understand the working of operating system services • Students will be able to write programs to use different OS APIs to access those services and write OS utility programs using them • Students will be able to write their own system services and add system calls inside the Kernel code	
Syllabus		
Contents	**Module-1 (Preparing your toolbox):** - Introduction - C-Compilation A system programmer perspective - Working of linkers and Creating your own libraries - UNIX make utility - GNU autotools and Cmake - Overview of versioning systems-git	

	- Exit Handlers - Process Stack behind the curtain - Process Heap behind the curtain **Module-2 (File, Information and Time Management):** - UNIX more utility - File system Architecture - File management in UNIX - Design and code of UNIX ls utility - Design and code of UNIX who utility - Programming the Terminals **Module-3 (Process Management and Scheduling):** - Process Management - Design and code of Daemon Service - Process Scheduling Algorithms - Design and code of UNIX shell - Thread Management **Module-4 (Inter-Process Communication):** - Overview of UNIX IPC and Signals on the Shell - Design and Code of Signal Handlers - Programming UNIX pipes - Programming UNIX named pipes - Message Queues - Programming with Shared Memory - Memory Mappings **Module-5 (Thread Management and Synchronization):** - Synchronization among Threads **Module-6 (Network Programming):** - Programming with POSIX semaphores - Overview of TCP/IP Architecture and Services - Socket Programming **Module-7 (Network Security):** - Vulnerabilities and exploits - Desiggning and injhecting Shell Code - Exploiting Buffer Overflow Vulnerability
Teaching-learning Strategies	• Lectures • Case Studies • Project • Assignments
Assignments	Types and Number with calendar
	Sr. # | Elements | Weightage | Details

Assessment and Examinations	1	Formative Assessment	25%	It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.
	2	Midterm Assessment	35%	It takes place at the mid-point of the semester.
	3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.
Textbooks	E. Advanced programming in the UNIX environment, by W. Richard Stevens; Stephen A. Rago, 4th edition, ISBN-13:9780321637734			
Reference Material/Suggested Readings	F. The Linux Programming Interface, by Michael Kerrisk, 2nd Edition, ISBN-13: 978-1593272203 G. Dr. Muhammad Arif Butt, System Programming - Video Lectures: https://www.youtube.com/c/LearnWithArif/playlists			
Notes				

Detailed Lecture wise plan

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
1	1	**Module-1 (Preparing your toolbox):** Discussion on Course Matrix. Application vs system programming perspective. How system calls work (Behind the curtain)	Text A-Ch1	
	2	**C-Compilation process** from system programmer perspective. Making a system call with and without wrapper. Making a system call from within Assembly program. (Tools used are gcc, gdb, readelf, objdump, nasm)	Text A-Ch1	
2	3	**Linking and loading** a process (Behind the curtain). Load time and run time dynamic linking. Structure of ELF file format. Merging re-locatable object files into executable. What is relocation. Symbol (global, external, local) resolution. Strong and weak symbols. Linker symbol rules for multiple strong and weak symbols. Creating and using your own static libraries. Creating and using your own dynamic libraries or shared objects. (Tools used are nm, ar, ranlib)	Text A-Ch2	
	4	**Introduction to make utility**, how make works, Makefile rules and targets. Using your own and built-in macros in a Makefile. Writing and calling multiple make files for a program. Comparison between binary and open source software packages. Downloading and installing open source software. Packaging your own software using GNU autotools (autoconf and automake). Packaging your own software using cmake utility. Writing your own man pages. (Tools used are make, autoconf, automake, cmake, cpack)	Text A-Ch2	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
3	5	**Local version control systems** (SCCS, RCS). Centralized version control systems (CVS, SVN). Distributed version control systems (git, bitkeeper, darcs, mercurial) Overview of git. Downloading and installing git. Basic git workflow and its configuration. Initializing git repository, Adding, editing, deleting and renaming files, and viewing commit log. Ignoring files in git and undoing changes. **Overview of git branches**. Creating, renaming, deleting and comparing branches. Overview of merging branches. The concept of fast forward and real merge. Handling merge conflicts. Concept of remote repositories and creating one on bitbucket. Pushing a local repo to bitbucket. Cloning an existing repo from bitbucket	Handouts	
	6	**How a C program starts** and terminates. Normal vs abnormal termination. Registering exit handlers using atexit() and on_exit() functions. Querying process resource limits and changing them on the shell. Getting and setting process resource limits from within a C program using getrlimit() and setrlimit() functions.	Text A-Ch7	**Lab:**
4	7	**Memory layout of a C program**. Use of command line arguments and environment variables in C programs. A discussion on layout of process stack and how it grows and shrinks. Stack buffer overflow problem. Doing a non-local goto using longjmp() and setjmp()	Text A Ch7	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
	8	**Allocating using and freeing memory on heap**. Layout of heap and heap allocators. Role of system calls brk() and sbrk() in management of heap. Common programming errors related to heap management. Tools and libraries for malloc debugging like splint, electric-fence, and valgrind	Text A-Ch7	**Lab:**
5	9	**Module-2 (File, Information and Time Management):** Hard disk partitioning, formatting and mounting. Internal structure of UNIX file system (Disk, partitions, boot block, super block, inode block). Connection to an open file via PPFDT. Universal I/O model (open-read-write-close paradigm). Kernel Buffering of file I/O and Buffering in the stdio library. Misc important file and directory related system calls	Ref A Ch3	
	10	**Different techniques of I/O redirection**. Use of fcntl() system call to duplicate a file descriptor, get/set file descriptor flags in PPFDT, get/set file status flags in SWFT, and achieving locks on files.	Text A-Ch3	
6	11	**Directory management in C Programs**. Working of UNIX ls utility. How it works? Can we design ls utility of our own? **Design and code of ls utility**	Ref A Ch4	
	12	**Design and code of uname and who utility**	Handout	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
7	13	**Module-3 (Process Management and Scheduling):** Process Identifiers: getpid(), getppid(), getuid(),setuid(), getgid(),setgid(), geteuid(),seteuid(), getegid(), setegid(). Process creation using fork(), vfork(), clone(). Copy on Write Semantics. Orphan and Zombie process	Text A Ch 6	
	14	**Process trees, chains and fans**. Monitoring Child Processes by accessing termination status of the child process using wait(), waitpid(), waitid(), wait3(), wait4() and WIFEXITED(), WEXITSTATUS(), WIFSIGNALED() and WTERMSIG() macros	Text A Ch 7	
8	15	**Six exec family functions** execl(), execle(), execlp(), execv(),execve(), execvp() Process Groups, Process Sessions, and concept of Controlling Terminals Impact of fork and exec on different process attributes. Writing your own system() function and use it to create a shell utility	Text A Ch 8	
9	16	**Overview of daemon processes** in Linux. Writing your own daemon process programatically. Introduction to systemd (replacement of SysV init daemon). Controlling daemons using systemctl utility. Writing long lived process and managing it using systemctl. **Design and Code your own daemon using systemd. Design and code of Shell utility (SPVL-22)**	Text A Ch 13	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
	17	**Different ways to achieve concurrency**. Conventional Concurrent Servers and their Problems. Thread implementation models. Linux implementation of POSIX threads (LinuxThreads and NPTL). Thread creation and termination, joining threads, passing implicit data types to thread function, and passing structures to thread function. Returning and receiving data from a thread function. Creating Thread arrays on stack, and creating thread arrays on heap. Thread attributes (detachstate, stackaddr, stacksize, priority, inheritsched). Changing the default attributes of a thread. Threads and signals. Threads and fork(). Thread cancellation.	Text A-Ch11	
10	18	**Module-4 (Inter-Process Communication):** Introduction to Linux Inter-Process communication tools. Overview of Standard and Real time Signals. Signal handling on Bash shell using kill and top utilities. Sending signals using kill, raise, abort, pause, alarm, sleep, usleep, nanosleep functions	Text A Ch 10	
	19	**Ignoring and handling signals** using signal() system call. Avoiding race conditions using signal mask and sigprocmask() system call. Limitations of signal() system call. Ignoring and handling signals using sigaction() system call. Scheduling future actions using interval timers getitimer(), setitimer(). Facts about standard I/O and redirection. I/O redirection using dup() and dup2()	Text A-Ch10	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
11	20	**Use of pipes** on the shell and in a C program using the pipe() system call. How to use a pipe for IPC among related processes for unidirectional communication. The concept of using two pipes for bidirectional communication. A C program that simulate the shell command man ls | grep ls | wc	Text A Ch 15	
	21	**Communication between unrelated processes using FIFOS** .Use of named pipes or FIFOs in a C program using the mkfifo() and mknod() system call. How to use a named pipe for IPC among un-related processes for unidirectional communication. The concept of using two named pipes for bidirectional communication. A simple client server application using FIFO.	Text A Ch 15	
12	22	**Module-5 (Thread Management and Synchronization):** Overview of synchronization. The concept of race condition and critical section problems. The concept of direct and indirect data sharing among threads, threads safety and reentrant functions. Introduction to mutex, locking, unlocking and destroying a mutex object. Mutex attributes and mutex types. Introduction to condition variable. The concept of thread cancellation is introduced.	Ref A Chapter 11	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
	23	**Module-6 (Network Programming):** Introduction to TCP/IP Programming Paradigm. Configuring some basic servers like telnet, dhcp, ssh, and ftp. Introduction to Socket, TLI, MacTCP and winsock Application Programming Interfaces. Introduction to TCP and UDP Client-Server Communication using BSD Socket API	Text A Ch 16	
13	24	**Flow chart showing TCP and UDP based client server** application using socket(), bind(), listen(), connect(), accept(), write(), send(), sendto(), read(), recv(), recvfrom(), and close().Three way connection establishment and four way connection termination.The related system calls for creating and managing the sockets.</br>Writing a helloworld tcp client server application	Text A Ch 16	
	25	**Proof of these concepts** with example codes of echo and daytime clients and servers. Some important lookup functions like gethostbyname() and getservbyname() with examples.Host vs NW Byte order (htons(), ntohs(), htonl(), ntohl()).	Text A Ch 16	
14	26	**Introduction to Datagram Sockets** by giving an overview of how datagram sockets work. The system call graph of Internet domain UDP Client and server. The related system calls for creating and managing the sockets. Proof of these concepts with example codes of echo, daytime and time clients and servers. Look up functions (gethostbyname(), getservbyname(), **Design and code of a web server**	Text A Ch 16	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
	27	**Module-7 (Network Security):** Overview of Cyber Security. Overview of buffer overflow. A classic stack based buffer overflow. How a stack based buffer overflow be exploited? Exploit mitigation techniques. The architecture of x86_64, its assembly and function calling convention. Installing and using PEDA. Changing the control of flow of execution in PEDA.	Handouts	
15	28	**Overview of shell code**. Writing your own shell code and using it in a stand alone C-program. Getting shell code from Internet archives and using them in a stand alone C-program.	Handouts	
	29	**Writing your shell codes** using pwn tools and using them in stand alone C-program. Writing your shell codes using msfvenom and using them in stand alone C-program	Handouts	
16	30	**Finding vulnerabilities in executables**. Crafting an input string to vulnerable programs to shift the control of flow of execution to some other part in the code section. Injecting shell code via input string and transferring control of flow to it. Injecting shell code via environment variables and transferring control of flow to it. Issues of exploiting vulnerable programs inside and outside gdb. Exploiting a vulnerable echo server executing on a remote machine and creating a tcp bind shell and a reverse tcp shell.	Handouts	

Program	BS Data Science	
Course Code	ED-443	
Course Title	Project Management & Quality Assurance	
Credit Hours	**Theory**	**Lab**
	3	0
Lecture Duration	90 minutes (1.5 Hours), 2 lectures per week	
Semester	8	
Pre-requisites	**Courses**	**Knowledge**
	Software Engineering	
Follow Up Courses		
Aims and Objectives	1. This course is aimed at introducing the primary important concepts of project management related to managing software development projects. 2. They will also get familiar with the different activities involved in Software Project Management. 3. Further, they will also come to know how to successfully plan and implement a software project management activity, and to complete a specific project in time with the available budget.	
Learning Outcomes	• CLO-1: Explain principles of the project lifecycle and how to identify opportunities to work with learners on relevant and appropriate project scenarios to share this understanding • CLO-2: Critically evaluate and discuss the issues around project management and its application in the real world with course participants and learners • CLO-3: Choose project management techniques for IT projects to initiate, plan, execute and evaluate a project and work in teams to create a project plan for a project scenario that includes key tasks, critical path, dependencies and a realistic timeline.	

<table>
<tr><td></td><td>CLO-4: Present strategies for gaining confidence in managing projects through simple project planning examples.</td></tr>
<tr><td>Syllabus</td><td>I. Introduction to Software Project Management, Project Management concepts, Project Management Tools, PMI's Knowledge areas, PMI Framework, PMI Process Groups. Understanding Organizations. Project Planning, Project Evaluation, Selection of an Appropriate Approach in Project, Software Effort Estimation, Activity Planning, Risk Management, Evaluating the Risks to the Schedule, Risk Control, Configuration Management and Maintenance, Environment for Configuration Control, Resource Allocation, Monitoring & Control, Review and Evaluation, Challenges of Outsourcing in Project Management</td></tr>
<tr><td>Contents</td><td>1. Introduction to software project management
2. Project evaluation and program management
3. An overview of project planning
4. Selection of an appropriate project approach
5. Software effort estimation
6. Activity planning
7. Risk management
8. Resource allocation
9. Monitoring and control
10. Managing contracts
11. Managing people in software environments
12. Working in teams
13. Software quality</td></tr>
<tr><td>Teaching-learning Strategies</td><td>The course will be based on the following teaching and learning activities:
Lectures covering the theoretical part using PowerPoint presentations
Case studies
Review questions</td></tr>
<tr><td>Assignments</td><td>Total 4 Assignment</td></tr>
<tr><td>Assessment and Examinations</td><td><table><tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr><tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>Assignments, Presentations, Quizzes.</td></tr></table></td></tr>
</table>

	2	Midterm Assessment	35%	Mid Term exam of 90 Minutes in 9th week of the semester.
	3	Final Assessment	40%	End Term exam of 120-180 minutes at the end of semester.
Textbooks	• Software Project Management, Bob Hughes and Mike Cotterell, McGraw-Hill Education;			
Reference Material/Suggested Readings	• Project Management Institute., & Project Management Institute,. (2013). A guide to the project management body of knowledge (PMBOK Guide). • Chemuturi, M., Cagley, T., & Safari, an O'Reilly Media Company. (2010). Mastering Software Project Management. • Wysocki, R. K. (2019). Effective Project Management: Traditional, Agile, Extreme. Newark: John Wiley & Sons, Incorporated. • Stellman, A., & Greene, J. (2008). Applied software project management. Sebastopol, CA: O'Reilly.			
Notes	• Power Point slides with reading material from book.			

Detailed Lecture wise plan

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
1	1	Project Management concepts, Project Management Tools, PMI's Knowledge areas, PMI Framework, PMI Process Groups.	PMBOK	
	2	Introduction to software project management Why is software project management important? What is a project? Software projects versus other types of project Contract management and technical project management Activities covered by software project management	Ch-01	
2	3	Plans, methods and methodologies Some ways of categorizing software projects Stakeholders Setting objectives The business case Project success and failure What is management? Management control	Ch-01	
	4	Project evaluation and program management A business case Project portfolio management Evaluation of individual projects Cost-benefit evaluation techniques	Ch-02	**Assignment-1**

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
3	5	Risk evaluation Program management Managing the allocation of resources within programs Strategic program management Creating a program Aids to program management Some reservations about program management Benefits management	Ch-02	**Quiz-1**
	6	An overview of project planning Introduction to Step Wise project planning Select project Identify project scope and objectives Identify project infrastructure Analyses project characteristics Identify project products and activities	Ch-03	
4	7	Estimate effort for each activity Identify activity risks Allocate resources Review/publicize plan Execute plan/lower levels of planning	Ch-03	
	8	Selection of an appropriate project approach Build or buy? Choosing methodologies and technologies Choice of process models Structure versus speed of delivery	Ch-04	
5	9	The waterfall model The spiral model Software prototyping	Ch-04	

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
	10	Other ways of categorizing prototypes Incremental delivery Agile methods Atern/Dynamic Systems Development Method	Ch-04	**Assignment-2**
6	11	Extreme programming (XP) Managing iterative processes Selecting the most appropriate process model	Ch-04	**Quiz-2**
	12	Software effort estimation Where are estimates done? Problems with over- and under-estimates The basis for software estimating Software effort estimation techniques Bottom-up estimating	Ch-05	
7	13	The top-down approach and parametric models Expert judgement Estimating by analogy Albrecht function point analysis Function points Mark II COSMIC full function points COCOMO 13: a parametric productivity model	Ch-05	
	14	Activity planning The objectives of activity planning When to plan Project schedules Projects and activities Sequencing and scheduling activities	Ch-06	

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
8	15	Network planning models Formulating a network model Adding the time dimension The forward pass The backward pass Identifying the critical path	Ch-06	
	16	Activity float Shortening the project duration Identifying critical activities Activity-on-arrow networks	Ch-06	
9	17	Risk Risk management Categories of risk A framework for dealing with risk	Ch-07	
	18	Risk identification Risk assessment Risk planning Risk management Evaluating risks to the schedule	Ch-07	**Assignment-3**
10	19	Applying the PERT technique Monte Carlo simulation Critical chain concepts	Ch-07	**Quiz-3**
	20	Resource allocation The nature of resources Identifying resource requirements Scheduling resources	Ch-08	
11	21	Creating critical paths Counting the cost Being specific Publishing the resource schedule Cost schedules The scheduling sequence	Ch-08	

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
	22	Monitoring and control Creating the framework Collecting the data Visualizing progress Cost monitoring	Ch-09	
12	23	Earned value analysis Prioritizing monitoring Getting the project back to target Change control	Ch-09	
	24	Managing contracts Types of contract Stages in contract placement	Ch-10	
13	25	Typical terms of a contract Contract management Acceptance	Ch-10	
	26	Managing people in software environments Understanding behavior Organization behavior: a background Selecting the right person for the job Instruction in the best methods	Ch-11	**Assignment-4**
14	27	Motivation The Oldham-Hackman job characteristics model Stress Health and safety Some ethical and professional concern	Ch-11	**Quiz-4**
	28	Working in teams Becoming a team Decision making Organizational structures Coordination dependencies	Ch-12	

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
15	29	Dispersed and virtual teams Communication genres Communication plans Leadership	Ch-12	
	30	Software quality The place of software quality in project planning The importance of software quality	Ch-13	
16	31	Defining software quality ISO 9126 Product versus process quality management Quality management systems Process capability models	Ch-13	
	32	Techniques to help enhance software quality Testing Quality plans	Ch-13	

<table>
<tr><td>Program</td><td colspan="2">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="2">MD-001</td></tr>
<tr><td>Course Title</td><td colspan="2">Math Deficiency I</td></tr>
<tr><td rowspan="2">Credit Hours</td><td>Theory</td><td>Lab</td></tr>
<tr><td>3</td><td>0</td></tr>
<tr><td>Lecture Duration</td><td colspan="2">90 minutes (1.5 Hours), 2 lectures per week</td></tr>
<tr><td>Semester</td><td colspan="2">1</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td>Courses</td><td>Knowledge</td></tr>
<tr><td>Nil</td><td>Nil</td></tr>
<tr><td>Follow Up Courses</td><td colspan="2"></td></tr>
<tr><td>Aims and Objectives</td><td colspan="2">1. Understand the basic concept of Complex numbers and its arithmetic properties
2. Learn about the idea of sequence and series, and their properties
3. Learn about Permutations and Combinations, Basic Probability
4. Understand the basic concept of Limits of functions, and its properties
5. Understand the basic concept of continuity and discontinuity of functions, and their properties
6. Understand the concept of derivatives, formulas and properties related to derivative
7. Under the concept of Increase, Decrease, Concavity, Relative Extrema, Absolute Maxima and Minima
8. Understand the Basic definitions of definite and indefinite Integrals,</td></tr>
</table>

	9. Learn about the Fundamental Theorem of Calculus 10. Learn how to Evaluate Definite Integrals by Substitution 11. Learn how to Evaluate the integral of Logarithmic and Other Functions
Learning Outcomes	• Students can understand what a computing problem is. • Students can formally define a computing problem. • Students can solve simple to moderate level computing problems. (Students can develop an I-P-O chart for a specific programming problem. Develop an algorithm to solve a specific programming problem by using pseudo-code or flowcharting. Use sequence, selection, and repetition structures to solve a problem. Apply the appropriate design for a specific program. Apply modular designs for programs, including sending arguments and returning values. Implement data validation, internal and external documentation in a program. Use objects and methods to solve problems in an object-oriented programming environment. Incorporate object-oriented tools within a program.)
Syllabus	Defining Set, various types of set representation and operations, Relation and function, Graphical transformation of one and two dimensional functions, Properties of functions, composition and inverses of functions, domain and range of the functions, Maximum and minimum values of functions, increasing and decreasing functions, zeros and intercept of functions, piecewise functions, continuity and Discontinuity of functions, Polynomials and rational functions, Polynomial long division and Synthetic division, Solution of rational functions, Absolute valued function, properties of absolute valued functions, Asymptotes (Horizontal, vertical and oblique), Exponential functions and their properties, Logs functions and their properties, Systems of Two Equations and Two Unknowns,

	Systems of Three Equations and Three Unknowns, Matrix Algebra (Add, subtract and multiply matrices), Row Operations and Row Echelon Forms, Augmented Matrices, Determinant of Matrices (2 x 2 and higher order matrices), Cramer's Rule, Inverse Matrices, Series and Sequences, Trigonometry, Angles in Radians and Degrees, Right Triangle Trigonometry, Law of Cosines & Sines, Area of Triangle, Graphs of Other Trigonometric Functions , Graphs of Inverse Trigonometric Functions, Basic Trigonometric Identities (Pythagorean, Sum and Difference, Double, Half, and Power Reducing), Trigonometric Equations, General Form of a Conic, Parabolas, Circles, Ellipses, Hyperbolas, Degenerate Conics, Polar and Parametric Equations, Polar and Rectangular Coordinates.
Contents	1. Sets 1.1. Defining Set, 1. various types of set representation and operations, 2. Relation and function, 2.1. Graphical transformation of one and two dimensional functions, 2.2. Properties of functions, 2.3. composition and inverses of functions 2.4. Domain and range of the functions 2.5. Maximum and minimum values of functions 2.6. Increasing and decreasing functions 2.7. Zeros and intercept of functions 2.8. Piecewise functions 2.9. Continuity and Discontinuity of functions, 3. Polynomials and rational functions 3.1. Polynomial long division and Synthetic division, 3.2. Solution of rational functions, 3.3. Absolute valued function, 3.4. properties of absolute valued functions, 3.5. Asymptotes (Horizontal, vertical and oblique), 3.6. Exponential functions and their properties, 3.7. Logs functions and their properties, 4. System of equations 4.1. Systems of Two Equations and Two Unknowns, 4.2. Systems of Three Equations and Three Unknowns, 5. Matrix Algebra (Add, subtract and multiply matrices), 5.1. Row Operations and Row Echelon Forms, 5.2. Augmented Matrices, Determinant of Matrices (2 x 2 and higher

<table>
<tr><td></td><td>order matrices),
5.3. Cramer’s Rule,
5.4. Inverse Matrices,
6. Series and Sequences,
7. Trigonometry,
7.1. Angles in Radians and Degrees,
7.2. Right Triangle Trigonometry,
7.3. Law of Cosines & Sines,
7.4. Area of Triangle,
7.5. Basic Trigonometric Identities (Pythagorean, Sum and Difference, Double, Half, and Power Reducing),
8. Graphs of Other Trigonometric Functions,
8.1. Graphs of Inverse Trigonometric Functions,
8.2. Trigonometric Equations,
9. General Form of a Conic,
9.1. Parabolas,
9.1.1. Circles, Ellipses, Hyperbolas,
9.1.2. Degenerate Conics,
9.1.3. Polar and Parametric Equations,
10. Polar and Rectangular Coordinates.</td></tr>
<tr><td>Teaching-learning Strategies</td><td>• Interactive class session
• Hands on practices in class
• Brainstorming and Group discussion sessions</td></tr>
<tr><td>Assignments</td><td>• Paper based written assignments</td></tr>
<tr><td>Assessment and Examinations</td><td><table>
<tr><th>Sr. #</th><th>Elements</th><th>Weightage</th><th>Details</th></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior, hands-on-activities, short tests, quizzes etc.</td></tr>
</table></td></tr>
</table>

	2	Midterm Assessment	35%	It takes place at the mid-point of the semester.
	3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.
Textbooks	• Textbook of Algebra and Trigonometry Class XI is published by Punjab Textbook Board (PTB) Lahore, Pakistan. • Calculus and Analytic Geometry, MATHEMATICS 12 (Mathematics FSc Part 2 or HSSC-II), Punjab Text Book Board Lahore, Pakistan			
Reference Material/Suggested Readings	• Gilbert, S. S., B. C. Andy and B. Andrew, B. 2005. Linear Algebra and Its Applications. 4th Ed. Thomson Brooks/Cole, Belmont, CA, USA. • Chung, S. K. 2014. Understanding basic calculus. Create Space Independent Publishing Platform, 173-175. • Howard, Anton, Irl Bivens, Stephen Davis, Calculus, 10^{th} Ed,2011, John Wiley & Sons, Inc. (1318 Pages) • https://www.maa.org/sites/default/files/images/upload_library/46/Pengelley_projects/Project-5/set_theory_project.pdf (An introduction to Elementary Set Theory by Guram Bezhanshvili and Eachan Landreth) • Howard Anton and Chris Rorres. Elementary linear Algebra, Wiley; 10th edition (April 12, 2010) • http://mecmath.net/trig/Trigonometry.pdf (Trigonometry Michael Corral)			
Notes	• Academic integrity is expected of all students. Plagiarism or cheating in any assessment will result in at least an F grade in the course, and possibly more severe penalties.			

	• You bear all the responsibility for protecting your assignments from plagiarism. If anyone else submits your assignment or uses your code in his/her assignment, you will be considered equally responsible. • The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester. • Introductory knowledge of using the computers is assumed for this course. All code written in quizzes, assignments, homework's, and exams must be in JavaScript. Code must be intelligently documented (commented). Undocumented code may not be given any credit. • The IDE use is not allowed, Notepad++ has to be used for coding. • There is no makeup for a missed sessional grading instruments like quizzes, assignments, and homework's.

Detailed Lecture wise plan

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
1	1	**Introduction to Sets:** Sets, subset relation, equality relation, set operations, set identities, Venn Diagrams, power sets.	R4	
	2	Russell's paradox, Cartesian products, relations, functions, one-to-one correspondences, functions equality, set equivalence.	R4	**Assign-1**
2	3	Cardinality of set, cardinal numbers, finite and infinite sets, countable sets, uncountable sets, independent and dependent variables in functions.	R4, R3(1)	**Quiz#1**
	4	Graphs of functions, vertical line test, the absolute value function and its properties.	R3(4)	
3	5	Piecewise defined functions, domains and ranges, the effect of algebraic operations on domain, domain and ranges in applied problems, issues of scale and units.	R3(6)	**Assign-2**
	6	Arithmetic operations on functions, composition of functions, translations, reflections, stretches and compressions, symmetry, even and odd functions.	R3(15)	**Quiz#2**
4	7	Polynomials, rational functions, algebraic functions, inverse functions, a method for finding inverse function, existence of inverse functions, graphs of inverse functions, restricting domains for invertibility.	R3(32, 38)	
	8	Exponential and logarithmic functions, solving equations involving exponentials and logarithms, change of base formula for logarithms, logarithmic scales in science and engineering, exponential and logarithmic growth.	R3(52)	
5	9	A brief review of polynomials, the remainder theorem, the factor theorem, synthetic division, using one factor to find other factors, methods for finding roots.	R3(A27)	
	10	**Matrices:** Linear equations, linear system with 2 and 3 unknowns, consistent and inconsistent systems, augmented matrix, elementary row operations.	R5(1.1)	**Assign-3**

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
6	11	Row echelon and reduced row echelon forms, Gaussian elimination, Gauss-Jordan elimination, free variables, one/two etc. parameter family of solution, homogenous linear systems, trivial and non-trivial solutions.	R5(1.2)	**Quiz#3**
	12	Formal definition of a matrix, operations on matrices, matrix product as linear combinations, transpose of a matrix, trace of a matrix.	R5(1.3)	
7	13	Properties of matrix arithmetic, Inverse of a matrix, singular and invertible matrices, inverse of 2×2 matrix, solution of a system of 2 equations in 2 unknowns by matrix inversion.	R5(1.4)	
	14	Method to find inverse of matrix by row operations, matrices those are not invertible, analyzing homogeneous systems, definition of determinants as a mapping, minors, cofactors, determinants by cofactor expansion, useful techniques to evaluate 2×2 and 3×3 determinants.	R5(1.5, 2.1)	**Assign-4**
8	15	Evaluating determinants by row reduction, properties of determinants.	R5(2.2, 2.3)	**Quiz#4**
	16	Adjoint of a matrix, inverse of a matrix by adjoint method, Cramer’s rule	R5(2.3)	
9	17	**Trigonometry:** Angles, types of triangles, Pythagorean theorem, Pythagorean triple, Euclid’s formula to generate Pythagorean triples.	R6(1)	
	18	Trigonometric functions of an acute angle, cofunction theorem, solving right angled triangles, applications of right angle triangles.	R6(7)	
10	19	Trigonometric functions of any angle, rotation and reflections of angles, general triangles, the law of sines.	R6(24)	**Assign-5**
	20	The law of cosines, the law of tangents, the area of a triangle, Heron’s formula.	R6(44)	**Quiz#5**
11	21	Circumscribed and inscribed circles, basic trigonometric identities, sum and difference formulas, double angle and half angle formulas.	R6(59)	
	22	Other identities, radians and degree, arc length, area of a sector.	R6(82)	

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
12	23	Graphing the trigonometric functions, properties of graphs of trigonometric functions, domains, ranges, periodicity.	R6(103)	**Assign-6**
	24	Inverse trigonometric functions, their domains and ranges, one-to-one correspondences, graphs of inverse trigonometric functions.	R6(120)	**Quiz#6**
13	25	Solving trigonometric equations, Polar Coordinates, relationship between polar and rectangular coordinates.	R6(129) R3(705)	
	26	Graphs in polar coordinates, symmetry tests, family of circles, family of rose curves, family of cardioids and limaçons, family of spirals.	R3(707)	
14	27	Conic sections; definitions of parabola, ellipse, and hyperbolas; directrix, focus (foci, plural), vertex and axis of symmetry, equations of parabolas in standard position, a technique for sketching parabolas.	R3(730)	**Assign-7**
	28	Equations of ellipses in standard position, a technique for sketching ellipses.	R3(734)	**Quiz#7**
15	29	Equations of hyperbolas in standard position, conjugate axis, asymptotes of hyperbolas, a technique for sketching hyperbolas.	R3(737)	
	30	Translated conics, Reflection properties of the conics, application of the conic sections.	R3(740)	
16	31	Rotation of axes; second degree equations, rotation of axes, eliminating the cross-product term.	R3(748)	**Assign-8**
	32	Conic sections in polar coordinates	R3(754)	**Quiz#8**

<table>
<tr><td>Program</td><td colspan="2">BS Data Science</td></tr>
<tr><td>Course Code</td><td colspan="2">MS-002</td></tr>
<tr><td>Course Title</td><td colspan="2">Math Deficiency II</td></tr>
<tr><td rowspan="2">Credit Hours</td><td>Theory</td><td>Lab</td></tr>
<tr><td>3</td><td>0</td></tr>
<tr><td>Lecture Duration</td><td colspan="2">90 minutes (1.5 Hours), 2 lectures per week</td></tr>
<tr><td>Semester</td><td colspan="2">2</td></tr>
<tr><td rowspan="2">Pre-requisites</td><td>Courses</td><td>Knowledge</td></tr>
<tr><td>Nil</td><td>Nil</td></tr>
<tr><td>Follow Up Courses</td><td colspan="2"></td></tr>
<tr><td>Aims and Objectives</td><td colspan="2">1. Understand the basic concept of Complex numbers and its arithmetic properties
2. Learn about the idea of sequence and series, and their properties
3. Learn about Permutations and Combinations, Basic Probability
4. Understand the basic concept of Limits of functions, and its properties
5. Understand the basic concept of continuity and discontinuity of functions, and their properties
6. Understand the concept of derivatives, formulas and properties related to derivative
7. Under the concept of Increase, Decrease, Concavity, Relative Extrema, Absolute Maxima and Minima
8. Understand the Basic definitions of definite and indefinite Integrals,
9. Learn about the Fundamental Theorem of Calculus</td></tr>
</table>

	10. Learn how to Evaluate Definite Integrals by Substitution 11. Learn how to Evaluate the integral of Logarithmic and Other Functions
Learning Outcomes	• Students can understand what a computing problem is. • Students can formally define a computing problem. • Students can solve simple to moderate level computing problems. (Students can develop an I-P-O chart for a specific programming problem. Develop an algorithm to solve a specific programming problem by using pseudo-code or flowcharting. Use sequence, selection, and repetition structures to solve a problem. Apply the appropriate design for a specific program. Apply modular designs for programs, including sending arguments and returning values. Implement data validation, internal and external documentation in a program. Use objects and methods to solve problems in an object-oriented programming environment. Incorporate object-oriented tools within a program.)
Syllabus	Complex Numbers, Arithmetic with Complex Numbers (Add, subtract, multiply and divide complex numbers), Trigonometric Polar Form of Complex Numbers, De Moivre's Theorem and nth Roots, Recursion, Arithmetic and Geometric Sequences, Sigma Notation, Arithmetic Series, Geometric Series (Sum infinite and finite geometric series and categorize geometric series), Counting with Permutations and Combinations, Basic Probability, Binomial Theorem, Limit Notation, Graphs to Find Limits, Tables to Find Limits, Substitution to Find Limits, Rationalization to Find Limits, One Sided Limits and Continuity, Instantaneous Rate of Change, Tangent Lines and Rates of Change, The Derivative Function, Introduction to Techniques of Differentiation, The Product and Quotient Rules, Derivatives of Trigonometric Functions, The Chain Rule, Derivatives of

	Logarithmic Functions, Derivatives of Exponential and Inverse Trigonometric Functions, Increase, Decrease, and Concavity, Relative Extrema, Absolute Maxima and Minima, An Overview of the Area Problem, Area Under a Curve, The Indefinite Integral, Integration by Substitution, The Definition of Area as a Limit; Sigma Notation, The Definite Integral.
Contents	**1.** Complex Numbers, 1.1. Arithmetic with Complex Numbers (Add, subtract, multiply and divide complex numbers), 1.2. Trigonometric Polar Form of Complex Numbers, 1.3. De Moivre’s Theorem and nth Roots, **2.** Recursion 2.1. Arithmetic and Geometric Sequences, 2.2. Sigma Notation, 2.3. Arithmetic Series, 2.4. Geometric Series (Sum infinite and finite geometric series and categorize geometric series), **3.** Counting 3.1. Permutations 3.2. Combinations, **4.** Basic Probability, **5.** Binomial Theorem, **6.** Limit Notation, 6.1. Graphs to Find Limits, 6.2. Tables to Find Limits, 6.3. Substitution to Find Limits, 6.4. Rationalization to Find Limits, 6.5. One Sided Limits and Continuity **7.** Instantaneous Rate of Change, 7.1. Tangent Lines and Rates of Change

<table>
<tr><td></td><td colspan="4">8. The Derivative Function,
8.1. Introduction to Techniques of Differentiation,
8.2. The Product and Quotient Rules,
8.3. Derivatives of Trigonometric Functions,
8.4. The Chain Rule,
8.5. Derivatives of Logarithmic Functions,
8.6. Derivatives of Exponential and Inverse
8.7. Trigonometric Functions,
9. Increase, Decrease, and Concavity,
9.1. Relative Extrema,
9.2. Absolute Maxima and Minima,
10. An Overview of the Area Problem,
10.1. Area Under a Curve,
10.2. The Indefinite Integral,
10.3. Integration by Substitution,
10.4. Sigma Notation,
11. The Definite Integral.
11.1. The Definition of Area as a Limit;</td></tr>
<tr><td>Teaching-learning Strategies</td><td colspan="4">• Interactive class session
• Hands on practices in class
• Brainstorming and Group discussion sessions</td></tr>
<tr><td>Assignments</td><td colspan="4">• Paper based written assignments</td></tr>
<tr><td rowspan="2">Assessment and Examinations</td><td>Sr. #</td><td>Elements</td><td>Weightage</td><td>Details</td></tr>
<tr><td>1</td><td>Formative Assessment</td><td>25%</td><td>It is continuous assessment. It includes: classroom participation, attendance, assignments and presentations, homework, attitude and behavior,</td></tr>
</table>

				hands-on-activities, short tests, quizzes etc.
	2	Midterm Assessment	35%	It takes place at the mid-point of the semester.
	3	Final Assessment	40%	It takes place at the end of the semester. It is mostly in the form of a test, but owing to the nature of the course the teacher may assess their students based on term paper, research proposal development, field work and report writing etc.
Textbooks	• Textbook of Algebra and Trigonometry Class XI is published by Punjab Textbook Board (PTB) Lahore, Pakistan. • Calculus and Analytic Geometry, MATHEMATICS 12 (Mathematics FSc Part 2 or HSSC-II), Punjab Text Book Board Lahore			
Reference Material/Suggested Readings	• Mark J. Christensen, Computing for Calculus, 1st Edition, Academic Press, (1st January 1981), 240pages, ISBN: 9781483271088. • Lay, L. D. 2015. Probability and Statistics for Engineering and the Sciences, 9th Ed. Cengage Learning, Boston, MA, USA. • Howard, Anton, Irl Bivens, Stephen Davis, Calculus, 10th Ed, 2011, John Wiley & Sons, Inc. (1318 Pages)			
Notes	• Academic integrity is expected of all students. Plagiarism or cheating in any assessment will result in at least an F grade in the course, and possibly more severe penalties. • You bear all the responsibility for protecting your assignments from plagiarism. If anyone else submits your assignment or uses your			

	code in his/her assignment, you will be considered equally responsible. • The instructor reserves the right to modify the grading scheme/marks division and course outline during the semester. • Introductory knowledge of using the computers is assumed for this course. All code written in quizzes, assignments, homework's, and exams must be in JavaScript. Code must be intelligently documented (commented). Undocumented code may not be given any credit. • The IDE use is not allowed, Notepad++ has to be used for coding. • There is no makeup for a missed sessional grading instruments like quizzes, assignments, and homework's.

Detailed Lecture wise plan

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
1	1	**Straight Lines:** Inclination of straight line, slope of lines, slope-intercept form of a line, two points form of a line.	A(18)	
	2	Intercept form of lines, normal form of lines, intersection of lines. Problem involving application of straight lines.	A(17)	**Assign-1**
2	3	**Complex Numbers:** Real number system, complex number system. The complex plane. Addition, multiplication of complex numbers. Properties of complex numbers. Inverse of a complex number.	A(17)	**Quiz#1**
	4	The modulus and conjugate of complex numbers. The argument of a complex number. The modulus-argument form of a complex number.	A(17)	
3	5	Multiplication and De Moivre's Theorem. Taking powers of a complex number.	A(17)	**Assign-2**
	6	Functions of a complex variable, the complex exponential function. Complex trigonometric function. Complex nth roots of a complex number.	B(7)	**Quiz#2**
4	7	**Permutations, Combinations and Binomial Theorem:** Counting principal and factorial, permutations, combinations, probability.	B(7)	
	8	Sequences. Sequences defined recursively. Arithmetic progression (A.P.). Common difference. General Term of A.P.	A(18)	
5	9	Geometric sequence. Common ratio. General term of a G.P. (Geometric progression).	A(18)	
	10	Series. Sum of arithmetic and geometric series involving finite terms. Sum of a convergent geometric series. Harmonic sequences.	A(18)	**Assign-3**
6	11	**Limits:** Limits an intuitive approach, one sided limit, two sided limits, sampling pitfalls, infinite limits, vertical asymptotes.	A(18)	**Quiz#3**

Week	Lecture	Topic	SourceBook (Ch#)	Recommendation for Learning Activities
	12	**Computing limits:** Some basic limits, limits of polynomials and rational functions as x approaches a, limits involving radicals, limits of piecewise functions.	B(18)	
7	13	**Limits at infinity:** Horizontal asymptotes, limits of rational functions when x approaches infinity, a quick method to evaluate a rational function when x approaches infinity, end behavior of trigonometric, logarithmic, exponential functions.	A(19) B(8)	
	14	Continuity of trigonometric, exponential and inverse functions, obtaining limits by squeezing.	Q	**Assign-4**
8	15	**Derivatives:** Tangent lines and rate of change, slopes, definition of derivative function, differentiability, the relationship between differentiability and continuity.	A(19) B(8)	**Quiz#4**
	16	Introduction to techniques of differentiation, derivative of a constant, derivatives of power functions, derivative of a constant times a function, derivatives of sum and differences, higher derivatives.	Hand outs	
9	17	The product and quotient rules, derivatives of trigonometric functions, the chain rule.	B(15) A(20)	
	18	**Implicit differentiation:** Functions defined explicitly and implicitly. Differentiability of functions defined implicitly. Derivatives of logarithmic functions. Logarithmic differentiation. Derivatives of real powers of x.	A(19) B(10)	
10	19	Derivatives of exponential and inverse trigonometric functions. Increasing or decreasing functions are one-to-one. Related rates. Local linear approximation.	A(18) B(15)	**Assign-5**
	20	**The derivative in graphing and applications:** Increasing and decreasing functions. Concavity. Inflection points. Logistic curves. Analysis of functions.	A(20) B(15)	**Quiz#5**

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
11	21	Critical points. Relative maxima and relative minima. First derivative test and second derivative test. Geometric implications of multiplicity. Analysis of polynomials.	B(14)	
	22	Absolute maxima and absolute minima. Applied maximum and minimum problems.	B(15)	
12	23	Rolle's theorem. Mean Value theorem. Velocity interpretation of Mean Value Theorem. Consequences of Mean Value Theorem. The constant difference theorem.	B(12)	**Assign-6**
	24	**Integration:** An overview of area problem. The rectangle problem of finding areas. The anti-derivative method for finding areas.	B(12)	**Quiz#6**
13	25	The indefinite integral. Integration formulas. Properties of indefinite integrals. Integration from the view point of differential equations. Slope fields.	B(12)	
	26	Integration by substitution. Easy to recognize substitutions. Less apparent substitutions. Integration using Computer Algebra Systems (CAS).	B(15)	
14	27	The definition of area as limit. Sigma notation. The Fundamental Theorem of Calculus. The relationship between definite and indefinite integrals. Differentiation and integration are inverse processes.	B(14)	**Assign-7**
	28	Evaluating definite integral by substitution. Area between curves.	A(20)	**Quiz#7**
15	29	Product rule and integration by parts. Guidelines for integration by parts. Repeated integration by parts. A tabular method for repeated integration by parts. Integration by parts for definite integrals.	A(20)	
	30	Reduction formulas. Integration of trigonometric functions. Wallis sine and cosine formulas.	A(20)	
16	31	Trigonometric substitutions. Integral involving $ax^2 + bx + c$. Integration of rational functions by partial fractions. Integrating improper rational functions. Improper integrals.	A(20)	**Assign-8**

Week	Lecture	Topic	Source Book (Ch#)	Recommendation for Learning Activities
	32	Taylors and Maclaurin Series. Approximations of functions in the vicinity of x=a.	Hand outs	**Quiz#8**