UNIVERSITY OF THE PUNJAB

NOTIFICATION

The Vice-Chancellor has, in exercise of the powers vested in him under Section 15(3) of the University of the Punjab Act, 1973 and in anticipation of approval of the Academic Council/Syndicate, been pleased to approve the recommendations of the Boards of Studies in Software Engineering, Information Technology and Computer Science and Board of Faculty of Computing & Information Technology at their meetings held on 10-04-2026, 23-04-2026, 30-04-2026 and 04-06-2026 respectively regarding the launch and approval of the Curriculum for the following Programs to be offered at University of the Punjab and its Affiliated Colleges from Fall 2026 :-

1. BS Computer Science
2. BS Computer Science (Specialization in Information Technology)
3. BS Computer Science (Specialization in Software Engineering)
4. BS Computer Science (Specialization in Artificial Intelligence)
5. Associate Degree in Computing
6. Associate Degree in Computing (Specialization in Information Technology)
7. Associate Degree in Computing (Specialization in Artificial Intelligence)

The Curriculum and Course Syllabi for above mentioned Programs are attached herewith, vide Annexure 'A' to 'G'.

Sd/-
Dr. Ahmad Islam
Registrar

Admin. Block,
Quaid-i-Azam Campus,
Lahore.
No. D/ 2690 /Acad. **Dated: 10-06-2026.**

Copy of the above is forwarded to the following for information and further necessary action: -

1. Pro-Chancellor/The Minister of Education, Govt. of the Punjab, Lahore.
2. Members of the Syndicate.
3. Dean, Faculty of Computing & Information Technology.
4. Chairman, Department of Computer Science/Software Engineering/ Information Technology
5. Principals of Affiliated Colleges (Concerned)
6. Controller of Examinations
7. Director. Quality Enhancement Cell
8. Director, IT (for Uploading on website)
9. Additional Registrar (Affiliation)
10. Deputy Registrar (General)
11. Secretary to the Vice-Chancellor
12. Secretary to the Pro-Vice-Chancellor
13. Secretary to the Registrar
14. Assistant Registrar (Statutes)
15. Assistant Registrar Syndicate (with file)
16. Assistant Registrar (Syllabus)

Dr. Ahmad Islam
Registrar

Program Curriculum

BS Computer Science

Department of Computer Science
University of the Punjab
Lahore

Curriculum, Course Syllabi, and Model Study Plan for BS Computer Science, prepared in accordance with the Revised Curriculum for Computer Science 2025 issued by the Higher Education Commission (HEC) of Pakistan (No. HEC/NCRC/CS&IT/2025/8163, dated October 14, 2025), for implementation at the University of the Punjab, Lahore, its sub-campuses, and its affiliated colleges from Fall 2026.

Programme	**BS Computer Science**				
Duration	**4 Years**	**Semesters**	**8**	**Credit hours**	**130**
Department	**Department of Computer Science**				
Faculty	**Faculty of Computing & Information Technology**				

Department Introduction

Welcome to the Department of Computer Science at the University of the Punjab, Lahore. As the oldest computing department in the university, we take pride in our rich history and legacy of excellence. Established in 1988 as the Center for Computer Science within the Centre for Solid State Physics at Quaid-e-Azam Campus, Lahore, the department quickly grew in stature. Just three years later, in 1991, we were elevated to the Department of Computer Science, becoming a key part of the Faculty of Engineering and Technology. By 2000, the department became part of the Faculty of Science and was upgraded to a Constituent College, named Punjab University College of Information Technology (PUCIT). In January 2021, with the creation of the Faculty of Computing and Information Technology (FCIT), PUCIT underwent restructuring into four departments, including the Department of Computer Science, to better cater to the evolving needs of the field. Our department has always been at the forefront of computer science education, offering Bachelor's, MPhil, and PhD degree programs. Over the years, we have proudly produced graduates at the BS, MPhil, and PhD levels who are making valuable contributions to academia, industry, and beyond. We offer a state-of-the-art curriculum that not only covers the fundamental concepts of computer science but also incorporates the latest advancements in machine learning, deep learning, and large language models. In addition, our graduates are equipped with strong software engineering skills, preparing them for the rapidly evolving tech landscape. The department is home to a dedicated and highly qualified faculty including more than 12 PhD holders, who bring expertise from diverse areas of computer science. Our degree programs are offered across two campuses: Hanjarwal Campus and the Allama Iqbal (Old Campus) of the University of the Punjab, Lahore. We are committed to continuing our tradition of excellence and innovation in the field of computer science, fostering an environment where students can thrive academically, professionally, and personally.

Department Vision

The vision of the Department of Computer Science, University of the Punjab, Lahore, is to impart computer science education at a global standard.

Department Mission

The mission of the department is to impart a strong foundation in the principles of computing, spanning from theoretical underpinnings to practical applications. We are committed to providing rigorous training in the systematic, efficient, and reliable development and maintenance of large-scale software systems. Through innovation and research, we strive to address socially relevant challenges using the power of computing. Our programs are designed to equip students with the skills and mindset needed to thrive in the rapidly evolving field of computer science, fostering lifelong learning, ethical responsibility, and leadership in the global digital society.

Program Education Objectives (PEOs)

The EOs of the BS Computer Science program are:

PEO 1	Provide students with a strong foundation in computing principles, from theoretical concepts to practical applications.
PEO 2	Provide rigorous training for systematic, efficient, and reliable development and maintenance of large software systems.
PEO 3	Innovate to address socially relevant challenges using computing.
PEO 4	Equip students to adapt to the fast-paced evolution of computer science, fostering lifelong learning and responsible leadership with effective communication.

The PEOs align with the department's vision of delivering a globally relevant computer science education and its mission to equip students with theoretical and practical knowledge, systematic software development skills, innovative problem-solving abilities, and leadership qualities. Each PO, focusing on technical expertise, communication, lifelong learning, and responsible leadership, meets institutional goals and stakeholder expectations for well-rounded, adaptable graduates.

Program Learning Outcomes (PLOs)

The Graduate Attributes (GAs) defined by the National Computing Education Accreditation Council (NCEAC) and adopted by the Department of Computer Science, University of the Punjab, are fully aligned with the Graduate Attributes prescribed in the Seoul Accord (Document D.5) for computing professionals. The range qualifier in several attribute statements incorporates the notions of complex computing problems and complex computing activities, which reflect the expected competence level of graduates. For the BS Computer Science program at the Department of Computer Science, University of the Punjab, Lahore, the Graduate Attributes (GAs) are directly adopted as Program Learning Outcomes (PLOs).

PLO #	GA #	Title	Description
PLO-1	GA-1	Academic Education	Completion of an accredited program of study designed to prepare graduates as computing professionals.
PLO-2	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.
PLO-3	GA-3	Problem Analysis	Identify and solve *complex* computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.
PLO-4	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.
PLO-5	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.
PLO-6	GA-6	Individual and Teamwork	Function effectively as an individual and as a member or leader of a team in multidisciplinary settings.
PLO-7	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.
PLO-8	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.
PLO-9	GA-9	Ethics	Understand and commit to professional ethics, responsibilities, and norms of professional computing practice.

PLO-10	GA-10	Life-long Learning	Recognize the need and have the ability to engage in independent learning for continual development as a computing professional.

PEO–PLO Mapping

The PEO–PLO mapping is presented in the following table, which shows how each Program Learning Outcome (PLO) contributes to achieving one or more Program Educational Objectives (PEOs).

PEOs / PLOs	PEO 1: Foundation in Computing Principles	PEO 2: Training for Large Software Systems	PEO 3: Innovation for Social Challenges	PEO 4: Lifelong Learning, Leadership & Communication
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				✔

Course-to-PLOs Mapping

Course-to-PLO mapping is a critical element of the Outcome-Based Education (OBE) framework because it establishes a clear traceability between what students learn in individual courses and the broader Program Learning Outcomes (PLOs) expected at graduation. This mapping ensures that all PLOs, which are derived from the Graduate Attributes defined by NCEAC, are adequately addressed across the curriculum, preventing overlaps or gaps in graduate competencies. It also provides measurable evidence of how course-level outcomes (CLOs) contribute to PLO attainment, which is essential for demonstrating compliance with NCEAC accreditation standards. Furthermore, by linking assessment data from courses to PLOs, departments can evaluate the extent to which desired outcomes are being achieved and implement continuous quality improvement (CQI) measures where needed. In this way, course-to-PLO mapping not only validates the alignment of the curriculum with the department's mission and Program Educational Objectives (PEOs) but also ensures that graduates develop a balanced set of technical, professional, and ethical skills required for their careers. The course-to-PLO mapping is presented in the following matrix.

Sr#	Course Code	Course Title	PLOs									
			1	2	3	4	5	6	7	8	9	10
1.	GE-160	Applications of Information & Communication Technologies	✓	✓	✓	✓	✓					
2.	GE-161	Discrete Structures	✓	✓	✓	✓						
3.	GE-162	Functional English	✓					✓	✓			✓
4.	GE-163	Introduction to Economics	✓	✓	✓					✓		
5.	GE-164	Probability & Statistics	✓	✓	✓	✓						
6.	GE-165	Expository Writing	✓					✓	✓			✓
7.	GE-166	Pakistan Studies	✓	✓	✓							
8.	GE-167	Fehm-e-Quran – I		✓						✓		✓
9.	GE-168	Fehm-e-Quran – II		✓						✓		✓
10.	GE-260	Civics and Community Management	✓		✓			✓	✓		✓	✓
11.	GE-261	Ideology and Constitution of Pakistan		✓						✓	✓	✓
12.	GE-262	Applied Physics	✓	✓	✓							
13.	GE-263	Professional Practices	✓							✓	✓	✓
14.	GE-264	Islamic Studies		✓					✓	✓	✓	✓
15.	GE-265	Entrepreneurship	✓	✓	✓	✓		✓	✓	✓		✓
16.	CSC-110	Programming Fundamentals	✓	✓	✓	✓	✓			✓	✓	✓
17.	CSC-110-L	Programming Fundamentals Lab	✓	✓	✓	✓	✓			✓	✓	✓
18.	CSC-111	Object Oriented Programming	✓	✓	✓	✓	✓	✓				
19.	CSC-111-L	Object Oriented Programming Lab	✓	✓	✓	✓	✓	✓	✓		✓	
20.	CSC-112	Digital Logic Design	✓	✓	✓					✓		✓
21.	CSC-112-L	Digital Logic Design Lab	✓	✓	✓					✓		✓
22.	CSC-210	Data Structures		✓	✓	✓						
23.	CSC-210-L	Data Structures Lab		✓	✓	✓						
24.	CSC-211	Database Systems		✓	✓	✓	✓			✓	✓	
25.	CSC-211-L	Database Systems Lab		✓	✓	✓	✓			✓	✓	
26.	CSC-212	Design & Analysis of Algorithms		✓	✓	✓						

27.	CSC-213	Operating Systems	✓	✓	✓	✓						
28.	CSC-213-L	Operating Systems Lab			✓	✓	✓	✓				
29.	CSC-214	Software Engineering			✓	✓	✓	✓	✓			✓
30.	CSC-215	Computer Organization & Architecture (Assembly language included)	✓	✓	✓	✓	✓					
31.	CSC-310	Computer Networks	✓	✓	✓	✓	✓					
32.	CSC-311	Artificial Intelligence	✓	✓	✓	✓	✓	✓	✓	✓	✓	
33.	CSC-312	Theory of Automata	✓	✓	✓	✓	✓					
34.	CSC-313	Information Security	✓	✓	✓	✓	✓			✓	✓	
35.	CSC-314	Cloud Computing	✓	✓	✓	✓	✓					
36.	CSE-320	Parallel & Distributed Computing	✓	✓	✓	✓	✓	✓	✓	✓	✓	
37.	CSE-321	Software Quality Engineering	✓			✓	✓	✓				
38.	CSE-322	Machine Learning	✓	✓	✓	✓	✓			✓	✓	
39.	CSE-323	Web Technologies		✓	✓	✓	✓	✓				✓
40.	CSE-324	Internet of Things and It's Applications	✓	✓	✓	✓	✓					
41.	CSE-325	Cryptography	✓	✓	✓	✓	✓					
42.	CSE-326	Compiler Construction	✓	✓	✓	✓	✓					
43.	CSE-421	Computer Vision	✓	✓	✓	✓	✓					
44.	CSE-422	Mobile Application Development	✓	✓	✓	✓	✓	✓				
45.	CSE-423	Human Computer Interaction	✓	✓	✓	✓	✓	✓	✓	✓	✓	
46.	CSE-424	Software Construction & Development		✓	✓	✓	✓	✓				
47.	CSE-425	Blockchain and Web 3 Security	✓	✓	✓	✓	✓	✓	✓	✓	✓	
48.	CSE-426	Big Data Analytics	✓		✓	✓	✓					
49.	CSE-427	Game Design and Development	✓	✓	✓	✓	✓	✓				
50.	CSE-428	DevOps Principles and Practices	✓	✓	✓		✓	✓		✓	✓	✓
51.	CSE-429	Low Code/No Code Application Development			✓	✓	✓			✓	✓	✓
52.	CSE-429-L	Low Code/No Code Application Development Lab			✓	✓	✓			✓	✓	✓
53.	CSI-130	Calculus and Analytical Geometry	✓		✓							
54.	CSI-230	Linear Algebra	✓	✓	✓	✓						
55.	CSI-330	Digital Marketing & E-Commerce	✓	✓	✓	✓		✓				
56.	CSI-331	Project Management	✓	✓	✓	✓	✓		✓			✓
57.	CSI-332	Bioinformatics	✓	✓	✓	✓						
58.	CSI-333	Geographic Information Systems (GIS) & Spatial Data	✓	✓	✓	✓						
59.	CSI-334	Cyber Law & Digital Policy		✓	✓	✓				✓	✓	✓
60.	CSP-450	Professional Certification	✓	✓	✓	✓	✓	✓				
61.	CSY-440	Final Year Project I	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
62.	CSY-441	Final Year Project II	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
63.	CSF-470	Internship	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Bloom's Taxonomy

Bloom's Taxonomy is a classification of the different outcomes and skills (PLOs). These 6 levels, shown in the table below, can be used to structure the learning outcomes, lessons, and assessments of a course. Each level of Bloom's taxonomy maps to one or more PLOs.

Code	Level	Description	Key Words
C1	**Remembering**	Can the students recall or remember the information?	know, describe, identify, label, list, match, memorize, recall
C2	**Understand**	Can the students explain ideas or concepts?	classify, describe, cite, discuss, generalize, illustrate, restate (in own words), summarize
C3	**Applying**	Can the students use the information in a new way?	assess, choose, solve, demonstrate, dramatize, establish, extend, illustrate,
C4	**Analyzing**	Can the students distinguish between different parts?	analyze, appraise, categorize, compare, identify, contrast, criticize, differentiate, recognize
C5	**Creating**	Can the students create new product or point of view?	adapt, incorporate, individualize, integrate, intervene, invent, model, modify, communicate, construct, create, design, develop, formulate, generate, reconstruct, reinforce,
C6	**Evaluating**	Can the students justify a stand or decision?	appraise, argue, choose, compare, conclude, contrast, criticize, interpret, judge, justify, predict, rate

Eligibility Criteria for Admission in BS Computer Science

- Higher Secondary School Certificate/A-levels (involving minimum 12 years of schooling) or an IBCC equivalent qualification, with at least 50% marks and having studied Mathematics, is the basic eligibility requirement for admission in all BS Computer Science.
    - The students who have not studied Mathematics at intermediate level have to pass deficiency courses of Mathematics (06 credits) in first year including zero semester.
- Additionally, applicants may be required to pass an entry test conducted by the department, university, or admission authority, along with fulfilling any other criteria set by the institution, such as interviews or aptitude assessments.

Program Structure of BS Computer Science

The degree program follows the HEC Undergraduate Education Policy 2023 V 1.1 and offers courses totaling 130 credit hours.

Total Credit Hours		**130**
General Education Courses		34 credit hours (14 courses)
Discipline Related Courses / Major	**Total**	**75 credit hours (26 courses)**
	Computer Science Core Courses	48 credit hours (14 courses)
	Specialization Electives	24 credit hours (08 courses)
	Professional Certification	03 credit hours
Interdisciplinary/ Allied Courses		12 credit hours (4 courses)
Internship		3 credit hours
Final Year Project		6 credit hours
Program Duration		Minimum: 4 Years Maximum: 6 Years (Further extendable to another year, subject to the approval of the university's statutory body following the provisions of HEC Undergraduate Education Policy 2023 V 1.1)
Semester Duration		16-18 weeks for regular semesters (1-2 weeks for examination) 8-9 weeks for summer semesters (1 week for examination)
Course Load (per semester)		15-21 credit hours for regular semesters Credit hours to be offered in Summer/Winter semesters, as per HEC guidelines (For remedial/deficiency/failure/ repetition courses only)
3 Credit Hours (Theory)		3 classes (1 hour each) OR 2 classes (1.5 hours each) OR 1 class (3 hours) per week throughout the semester.
1 Credit Hours (Practical Work)*		1 Credit hour of practical work requires three contact hours per week throughout the semester.

Nomenclature of Course Categories and Course Codes

N	Code	Description	Courses	Cr. Hrs. (Theory, Lab)	
0	MD	Math Deficiency	2*	6*	(6,0)*
1	CSC	Computer Science Core Courses	14	48	(41,7)
2	CSE	Computer Science Elective Courses	8	24	(24,0)
3	CSI	Interdisciplinary/Allied Courses	4	12	(12,0)
4	CSY	Final Year Project	-	6	(0,6)
5	CSP	Professional Certification	-	3	(0,3)
6	GE	General Education Courses	14	34	(30,4)
7	CSF	Internship	-	3	(0,3)
		Total	**42**	**130**	**(107,23)**

Course Coding Scheme

Code-YNS

Code = MD, CSC, CSE, GE, CSI, CSP, CSF, CSY

Y = Year of Offering (Earliest) = 1, 2, 3, 4

N = Numeric Code = 0, 1, …, 8

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

-L = A lab course code, same as related theory course codes with "-L" as a suffix

*Two non-credited math deficiency courses to be offered to the students who have not studied mathematics in their Intermediate, A-level or equivalent.

List of Courses

GENERAL EDUCATION: 34 (30, 4)

Sr.	Code	Course Title	Sub – Category	Cr. Hrs.
1.	GE-160	Applications of Information & Communication Technologies		3 (2,1)
2.	GE-161	Discrete Structures	Quantitative Reasoning – I	3 (3,0)
3.	GE-162	Functional English		3 (3,0)
4.	GE-163	Introduction to Economics	Social Science	2 (2,0)
5.	GE-164	Probability & Statistics	Quantitative Reasoning – II	3 (3,0)
6.	GE-165	Expository Writing		3 (3,0)
7.	GE-166	Pakistan Studies		2 (2,0)
8.	GE-167	Fehm-e-Quran – I		1 (0,1)
9.	GE-168	Fehm-e-Quran – II		1 (0,1)
10.	GE-260	Civics and Community Management		2 (2,0)
11.	GE-261	Ideology and Constitution of Pakistan		2 (2,0)
12.	GE-262	Applied Physics	Natural Science	3 (2,1)
13.	GE-263	Professional Practices	Arts and Humanities	2 (2,0)
14.	GE-264	Islamic Studies		2 (2,0)
15	GE-265	Entrepreneurship		2 (2,0)

MATHEMATICS DEFICIENCY: 6 (6,0)

Sr.	Code	Course Title	Prerequisite	Cr. Hrs.
1.	MD-001	Math Deficiency - I		3 (3,0)*
2.	MD-002	Math Deficiency - II		3 (3,0)*

COMPUTER SCIENCE CORE: 48 (41,7)

Sr.	Code	Course Title	Prerequisite	Cr. Hrs.
1.	CSC-110	Programming Fundamentals		3 (3,0)
2.	CSC-110-L	Programming Fundamentals Lab		1 (0,1)
3.	CSC-111	Object Oriented Programming	CSC-110	3 (3,0)
4.	CSC-111-L	Object Oriented Programming Lab	CSC-110	1 (0,1)
5.	CSC-112	Digital Logic Design		3 (3,0)
6.	CSC-112-L	Digital Logic Design Lab		1 (0,1)
7.	CSC-210	Data Structures	CSC-111	3 (3,0)
8.	CSC-210-L	Data Structures Lab	CSC-111	1 (0,1)
9.	CSC-211	Database Systems		3 (3,0)
10.	CSC-211-L	Database Systems Lab		1 (0,1)
11.	CSC-212	Design & Analysis of Algorithms	CSC-210	3 (3,0)
12.	CSC-213	Operating Systems		3 (3,0)
13.	CSC-213-L	Operating Systems Lab		1 (0,1)
14.	CSC-214	Software Engineering		3 (3,0)
15.	CSC-215	Computer Organization & Architecture (Assembly language included)	CSC-112	3 (2,1)
16.	CSC-310	Computer Networks		3 (3,0)
17.	CSC-311	Artificial Intelligence		3 (3,0)
18.	CSC-312	Theory of Automata		3 (3,0)
19.	CSC-313	Information Security		3 (3,0)
20.	CSC-314	Cloud Computing		3 (3,0)

COMPUTER SCIENCE ELECTIVE: 24 (24, 0)

Sr.	Code	Course Title	Prerequisite	Cr. Hrs.
1.	CSE-320	Parallel & Distributed Computing	CSC-213	3 (3,0)
2.	CSE-321	Software Quality Engineering	CSC-214	3 (3,0)
3.	CSE-322	Machine Learning	CSI-230, GE-164	3 (3,0)
4.	CSE-323	Web Technologies	CSC-210	3 (3,0)
5.	CSE-324	Internet of Things and It's Applications		3 (3,0)
6.	CSE-325	Cryptography		3 (3,0)
7.	CSE-326	Compiler Construction	CSC-312	3 (3,0)
8.	CSE-421	Computer Vision	CSI-230, GE-164	3 (3,0)
9.	CSE-422	Mobile Application Development	CSC-210	3 (3,0)
10.	CSE-423	Human Computer Interaction		3 (3,0)
11.	CSE-424	Software Construction & Development	CSC-210	3 (3,0)
12.	CSE-425	Blockchain and Web 3 Security		3 (3,0)
13.	CSE-426	Big Data Analytics		3 (3,0)
14.	CSE-427	Game Design and Development	CSC-210	3 (3,0)
15.	CSE-428	DevOps Principles and Practices	CSC-210	3 (3,0)
16.	CSE-429	Low-code/No-code Application Development		2 (2,0)
17.	CSE-429-L	Low-code/No-code Application Development Lab		1 (0,1)

Interdisciplinary / Allied Courses: 12 (12, 0)

Sr.	Code	Course Title	Prerequisite	Cr. Hrs.
1.	CSI-130	Calculus and Analytical Geometry (Mandatory)		3 (3,0)
2.	CSI-230	Linear Algebra (Mandatory)		3 (3,0)
3.	CSI-330	Digital Marketing & E-Commerce		3 (3,0)
4.	CSI-331	Project Management		3 (3,0)
5.	CSI-332	Bioinformatics		3 (3,0)
6.	CSI-333	Geographic Information Systems (GIS) & Spatial Data		3 (3,0)
7.	CSI-334	Cyber Law & Digital Policy		3 (3,0)

Scheme of Studies / Semester-wise workload

Semester - I						
Sr.	**Code**	**Course Title**	**Pre-Requisite**	**Domain**	**Cr. Hrs. (T, L)**	
1.	GE-160	Applications of Information & Communication Technologies		General Education	3	(2,1)
2.	CSC-110	Programming Fundamentals		CS Core	3	(3,0)
3.	CSC-110-L	Programming Fundamentals Lab		CS Core	1	(0,1)
4.	GE-161	Discrete Structures		General Education	3	(3,0)
5.	GE-162	Functional English		General Education	3	(3,0)
6.	GE-163	Introduction to Economics		General Education	2	(2,0)
7.	GE-166	Pakistan Studies		General Education	2	(2,0)
8.	GE-167	Fehm-e-Quran – I		General Education	1	(0,1)
9.	MD-001	Math Deficiency – I		Math Deficiency	3*	(3,0)*
				Credit Hours (Semester – I)	**18**	**(15,3)**

Semester – II						
Sr.	**Code**	**Course Title**	**Pre-Requisite**	**Domain**	**Cr. Hrs.**	
1.	CSC-111	Object Oriented Programming	CSC-110	CS Core	3	(3, 0)
2.	CSC-111-L	Object Oriented Programming Lab	CSC-110	CS Core	1	(0,1)
3.	CSC-112	Digital Logic Design		CS Core	3	(3,0)
4.	CSC-112-L	Digital Logic Design Lab		CS Core	1	(0,1)
5.	GE-164	Probability & Statistics		General Education	3	(3,0)
6.	GE-165	Expository Writing		General Education	3	(3,0)
7.	GE-168	Fehm-e-Quran – II		General Education	1	(0,1)
8.	CSI-130	Calculus and Analytical Geometry		IDS - Mandatory	3	(3,0)
9.	MD-002	Math Deficiency – II		Math Deficiency	3*	(3,0)*
				Credit Hours (Semester – II)	**18**	**(16,3)**

Semester – III						
Sr.	**Code**	**Course Title**	**Pre-Requisite**		**Cr. Hrs.**	
1.	CSC-210	Data Structures	CSC-111	CS Core	3	(3,0)
2.	CSC-210-L	Data Structures Lab	CSC-111	CS Core	1	(0,1)
3.	CSC-215	Computer Organization & Architecture	CSC-112	CS Core	3	(2,1)
4.	GE-260	Civics and Community Engagement		General Education	2	(2,0)
5.	GE-261	Ideology and Constitution of Pakistan		General Education	2	(2,0)
6.	GE-264	Islamic Studies (Religious Education / Ethics for non-Muslim students)		General Education	2	(2,0)
7.	GE-265	Entrepreneurship		General Education	2	(2,0)
8.	CSI-230	Linear Algebra		IDS - Mandatory	3	(3,0)
				Credit Hours (Semester – III)	**18**	**(16,2)**

Semester – IV						
Sr.	**Code**	**Course Title**	**Pre-Requisite**	**Domain**	**Cr. Hrs.**	
1.	CSC-211	Database Systems		CS Core	3	(3, 0)
2.	CSC-211-L	Database Systems Lab		CS Core	1	(0, 1)
3.	CSC-212	Design & Analysis of Algorithms	CSC-210	CS Core	3	(3,0)
4.	CSC-213	Operating Systems		CS Core	3	(3,0)
5.	CSC-213-L	Operating Systems Lab		CS Core	1	(0,1)
6.	CSC-214	Software Engineering		CS Core	3	(3,0)
7.	GE-263	Professional Practices		General Education	2	(2,0)
8.	GE-262	Applied Physics		General Education	3	(2,1)
				Credit Hours (Semester – IV)	**19**	**(16,3)**

Semester – V						
Sr.	**Code**	**Course Title**	**Pre-Requisite**	**Domain**	**Cr. Hrs.**	
1.	CSC-310	Computer Networks		CS Core	3	(3,0)
2.	CSC-311	Artificial Intelligence	CSI-230, GE-164	CS Core	3	(3,0)
3.	CSC-312	Theory of Automata		CS Core	3	(3,0)
4.	CSE-323	Web Technologies / Elective I	CSC-210	CS Elective	3	(3,0)
5.	CSI-330	Digital Marketing & E-Commerce		IDS	3	(3,0)
6.	CSI-331	Project Management		IDS	3	(3,0)
				Credit Hours (Semester – V)	**18**	**(18,0)**

Semester – VI						
Sr.	**Code**	**Course Title**	**Pre-Requisite**	**Domain**	**Cr. Hrs.**	
1.	CSC-313	Information Security		CS Core	3	(3,0)
2.	CSC-314	Cloud Computing		CS Core	3	(3,0)
3.	CSE-320	Parallel & Distributed Computing / Elective II	CSC-214	CS Elective	3	(3,0)
4.	CSE-321	Software Quality Engineering / Elective III	CSC-214	CS Elective	3	(3,0)
5.	CSE-322	Machine Learning / Elective IV	CSI-230, GE-164	CS Elective	3	(3,0)
				Credit Hours (Semester – VI)	**15**	**(15,0)**

Semester – VII						
Sr.	**Code**	**Course Title**	**Pre-Requisite**	**Domain**	**Cr. Hrs.**	
1.	CSE-420	Software Construction & Development / Elective V	CSC-210	CS Elective	3	(3,0)
2.	CSE-421	Computer Vision / Elective VI	CSI-230, GE-164	CS Elective	3	(3,0)
3.	CSE-422	Mobile Application Development / Elective VII	CSC-210	CS Elective	3	(3,0)
4.	CSE-428	Game Design and Development / Elective VIII		CS Elective	3	(3,0)
5.	CSY-440	Final Year Project – I		Final Year Project	3	(0,3)
					15	**(12,3)**

Semester – VIII						
Sr.	**Code**	**Course Title**	**Pre-Requisite**	**Domain**	**Cr. Hrs.**	
1.	CSY-441	Final Year Project – II	CSY-440	Final Year Project	3	(0,3)
2.	CSP-450	Professional Certification		Professional Certification	3	(0,3)
3.	CSF-470	Internship		Internship	3	(0,3)
				Credit Hours (Semester – VIII)	**9**	**(0,9)**
				Total Credit Hours	**130**	**(107,23)**

Professional Certifications: 3 (0,3)

- Students of BS Computer Science are required to complete the international certification(s) equivalent to 3 credit hours with at least 48 hours of engagement. The student can take one or more certifications to fulfill the requirement of at least 48 hours of engagement.
- BS Computer Science students are required to complete professional certification(s) (equivalent to 3 credit hours in total) over the period of the four-year program as a mandatory condition for degree completion. A certification will be considered equivalent to 1 credit hour if it comprises at least 16 hours of study.
- The university/department may verify the validity (not expired) and authenticity of the certification with the issuing body and conduct a university-administered exam/viva/presentation to award the marks/grade of 3 credit hours of Professional Certifications.

Approved Certifications:

The respective department will guide students in selecting relevant certifications, ensuring alignment with current market needs and the program's objectives.

- Students can choose from the list of certifications recommended by Pakistan Software Export Board (PSEB), Ministry of Information Technology and Telecommunication (MoITT) available at the website: https://techdestination.com/.
- Students can also choose certifications from well-known international certification organizations/ platforms, including edX, Coursera, Domestika, DataCamp, FutureLearn, IBM, Microsoft Learn, Skillshare, Udacity, and Udemy.

Field Experience/Internship: 3 (0,3)

- Students of BS Computer Science are required to complete the Internship equivalent to 3 credit hours with at least 150 work hours.
- Students must complete their internship at any time during the 4-year degree. However, it is recommended to do it after the 2nd year (i.e., after completing 4 semesters).
- Internships of three (03) credit hours must be completed in accordance with HEC Undergraduate Education Policy V1.1. This requirement cannot be substituted with additional coursework, capstone, research, or project work.
- Internships can be completed through any of the following modes:
    - On-site (Physical) – Working at an organization's office or workplace.
    - Remote / Online – Working virtually for an organization.
    - University-assigned Projects/Tasks – Completing approved work given by the university.
    - Industry-focused Courses / Workshops / Bootcamps offered or approved by the university.
- The university/department may verify the authenticity of the internship and establish a proper evaluation mechanism to assess the internship.
- One or more of the following instruments may be used to evaluate the internship and award marks/grades.
    - Internship reports/logbooks
    - Employer/supervisor feedback
    - Presentations or viva voce
    - Project deliverables

Course Title	**Internship**		
Course Code	**CSF-470**		
Credit Hours	3 (0,3)		
Category	Internship		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	The Internship is a practical training component of the BS Computer Science program designed to provide students with real-world exposure to professional computing environments. It enables students to apply their academic knowledge and technical skills in industry settings while gaining experience in teamwork, communication, and workplace practices. Through supervised industry placement, students engage in problem-solving, use modern tools and technologies, and understand organizational workflows. The internship also promotes professional ethics, responsibility, and self-directed learning, helping students bridge the gap between theory and practice and preparing them for successful careers in the computing field.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO-1: Apply computing knowledge and technical skills to perform assigned tasks in a professional environment.	C3 (Apply)	1,2,4
	CLO2: Solve the assigned tasks on time by using different modes of communication.	C2 (Understand)	2,3
	CLO3: Apply the key findings to improve their assigned tasks, such as	C3 (Apply)	4,5

	assignments, projects, and presentations.		
	CLO4: Analyze the complete SDLC phases of any working product.	C4 (Analyze)	3,4,5
	CLO5: Develop the skills that help to solve complex problems for real-time applications.	C5 (Design)	6,7,10
	CLO6: Demonstrate professional ethics, responsibility, and awareness of workplace norms and societal impact	C3 (Apply)	8,9

Final Year Project (FYP): 6 (0,6)

- Students of BS Computer Science are required to complete the Final Year Project of 6 credit hours.
- All projects should be aligned with one or more UN Sustainable Development Goals (SDGs) or relevant thematic areas (e.g., health, education, environment, smart cities, innovation, etc.).
- Each project must be supervised by a faculty member approved by the department. Co-supervisors from industry are encouraged where applicable.
- Students may work individually or in teams (as per department policy), but the workload should match the credit hours.
- Each student must have a clearly defined role.

The BSCS Final Year Project (FYP) is assessed through multiple stages, including proposal approval, mid-term progress evaluations, final presentation or demonstration, report submission, and, where applicable, external evaluation to ensure quality.

Upon completion, students must submit essential deliverables such as a final report or thesis, presentation slides, a working prototype or system, and a poster or summary.

Throughout the project, students are required to follow ethical and professional standards by avoiding plagiarism and properly acknowledging any use of licensed data, copyrighted materials, or AI tools.

Course Title	**Final Year Project**		
Course Code	**CSY-440, CSY-441**		
Credit Hours	6 (0,6)		
Category	Final Year Project		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	The Final Year Project (FYP) is a capstone experience designed to integrate and apply the knowledge, skills, and competencies acquired throughout the BS Computer Science program. It provides students with the opportunity to identify and address a real-world computing problem through systematic research, design, development, and evaluation of a complete solution. The project emphasizes independent learning, critical thinking, innovation, and problem-solving, while also fostering skills in teamwork, project management, and technical communication. Students are expected to use appropriate modern tools and technologies, follow ethical and professional practices, and consider societal and legal implications of their work. The FYP serves as a key indicator of a student's readiness to enter the computing profession or pursue advanced studies, and it plays a central role in demonstrating the attainment of Program Learning Outcomes (PLOs) within an Outcome-Based Education (OBE) framework.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Identify the key concept of the project domain. Identify, formulate, and analyze a real-world computing problem through a literature review.	C4 (Analyze)	1,2,3

	CLO2: Implement the best suitable framework, architecture, algorithm, libraries, and techniques for the selected domain	C3 (Apply)	3,4,5
	CLO3: Use key practices defined by the UN Sustainable Development Goals (SDGs).	C3 (Apply)	2,3
	CLO4: Analyze the proposed approach based on current practices defined by the IEEE/ ISO/ IETF for a selected problem domain that helps to launch a working product at global platform.	C4 (Analyze)	3,4,5
	CLO5: Build a working product that covers the complete functional as well as non-functional requirements by semi or fully automated approaches using programming, and analyzing data, ensuring the system meets specified project requirements.	C5 (Design)	6,7
	CLO6: Evaluate the proposed approach based on the key metrics for the selected domain and compare the findings based on available datasets	C6 (Evaluate)	8,9
	CLO7: Demonstrate awareness of ethical, legal, and societal implications of the proposed solution	C3 (Apply)	8,9,10

MATHEMATICS DEFICIENCY: 6 (6,0)

Sr.	Code	Course Title	Prerequisite	Cr. Hrs.
1.	MD-001	Math Deficiency - I		3 (3,0)*
2.	MD-002	Math Deficiency - II		3 (3,0)*

Course Title	**Math Deficiency – I**		
Course Code	**MD-001**		
Credit Hours	3*		
Category	Mathematics Deficiency		
Prerequisite	None		
Co-Requisite	None		
Follow Up	Math Deficiency – II		
Course Introduction	This course introduces fundamental concepts in algebra, functions, and trigonometry, providing a strong base for further study in mathematics. It covers sets and their operations, relations and functions with their properties and graphs, and techniques for solving systems of equations using algebraic and matrix methods. Students explore sequences, series, and various types of functions, including polynomial, rational, exponential, and logarithmic forms. The course also includes trigonometry, conic sections, and polar and parametric equations, helping students develop analytical and problem-solving skills essential for advanced mathematical applications.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Know the concepts and applications of sets, relations, functions, systems of equations, trigonometric functions and matrix algebra	C1 (Know)	1
	CLO2: Describe system of linear equations, matrix algebra, trigonometry, and related techniques.	C2 (Describe)	1
	CLO3: Solve problems related to system of linear equations, matrix algebra, and trigonometry.	C3 (Apply)	1,3
	CLO4: Know the general form of Conic, polar coordinate and parametric equations.	C1 (Know)	1
Text Book(s)	1. Textbook of Algebra and Trigonometry Class XI is published by Punjab Textbook Board (PTB) Lahore, Pakistan. 2. Calculus and Analytic Geometry, MATHEMATICS 12 (Mathematics FSc Part 2 or HSSC-II), Punjab Text Book Board Lahore, Pakistan		
Reference Material	1. Gilbert, S. S., B. C. Andy and B. Andrew, B. 2005. Linear Algebra and Its Applications. 4th Ed. Thomson Brooks/Cole, Belmont, CA, USA.		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Definition of sets Types of sets (finite, infinite, empty, singleton, universal) Set representation (roster & set-builder form)	Book1: Chapter 2
Week 2	Set operations: union, intersection, difference, complement Venn diagrams Basic applications and word problems	Book1: Chapter 2 Quiz 1
Week 3	Relations and functions (definitions) Domain and range Types of functions Graphing basic functions	Thomas Jr. et al.: Chapter 1 Assignment 1
Week 4	Function transformations (shifts, reflections, stretches) Composition of functions Inverse functions	Thomas Jr. et al.: Chapter 1
Week 5	Increasing/decreasing functions Maximum and minimum values Zeros and intercepts Piecewise functions	Thomas Jr. et al.: Chapter 1
Week 6	Continuity and discontinuity	Thomas Jr. et al.: Chapter 1

	Polynomial functions (intro) Rational functions (intro)	
Week 7	Polynomial long division Synthetic division Solving rational functions Graphing rational functions	Lecture handouts Assignment 2
Week 8	Absolute value functions & properties Asymptotes (horizontal, vertical, oblique) Exponential functions and properties Logarithmic functions and properties	Thomas Jr. et al.: Chapter 2 Quiz 2
Week 9	Systems of two equations (graphical & algebraic methods) Systems of three equations	Book1: Chapter 3
Week 10	Matrix algebra (addition, subtraction, multiplication) Row operations and row echelon form Augmented matrices	Book1: Chapter 3
Week 11	Determinants & Inverses Determinants (2×2 and higher order) Cramer's Rule Inverse matrices Solving systems using matrices	Book1: Chapter 3 Assignment 3
Week 12	Sequences & Series Introduction to sequences Arithmetic and geometric sequences Series basics	Book1: Chapter 6 Quiz 3
Week 13	Angles (degrees & radians) Right triangle trigonometry Law of Sines and Cosines Area of triangle	Thomas Jr. et al.: Chapter 1
Week 14	Graphs of trigonometric functions Inverse trigonometric functions Basic identities (Pythagorean, sum/difference, double/half angle)	Thomas Jr. et al.: Chapter 1 Quiz 4
Week 15	Trigonometric Equations & Conics Solving trigonometric equations General form of conics Parabolas Circles	Book 2: Chapter 6 Assignment 4
Week 16	Ellipses Hyperbolas Polar and rectangular coordinates Parametric equations	Book 2: Chapter 6

Course Title	**Math Deficiency – II**		
Course Code	**MD-002**		
Credit Hours	3*		
Category	Mathematics & Supporting (Deficiency Course)		
Prerequisite	None		
Co-Requisite	None		
Follow Up	CSI-130 Calculus & Analytic Geometry		
Course Introduction	This course introduces key concepts of complex numbers, sequences and series, counting techniques, probability, and the binomial theorem. It then develops foundational ideas of calculus, including limits, continuity, rates of change, derivatives, and their applications in analyzing functions. The course concludes with an introduction to integrals and area under curves, equipping students with essential mathematical tools for further studies.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Know the concepts and applications of complex number, sequences, series, permutations and combinations, integration and differentiation	C1 (Know)	1
	CLO2: Describe functions, limit, continuity chain rule and related techniques.	C2 (Describe)	1
	CLO3: Identify and solve problems related to differentiation and integration.	C3 (Apply)	1,3
Text Book(s)	1. Textbook of Algebra and Trigonometry Class XI is published by Punjab Textbook Board (PTB) Lahore, Pakistan. 2. Calculus and Analytic Geometry, MATHEMATICS 12 (Mathematics FSc Part 2 or HSSC-II), Punjab Text Book Board Lahore, Pakistan		
Reference Material	1. Mark J. Christensen, Computing for Calculus, 1st Edition, Academic Press, (1st January 1981), 240 pages, ISBN: 9781483271088. 2. Lay, L. D. 2015. Probability and Statistics for Engineering and the Sciences, 9th Ed. Cengage Learning, Boston, MA, USA. 3. Howard, Anton, Irl Bivens, Stephen Davis, Calculus, 11th Ed, 2011, John Wiley & Sons, Inc. (1318 Pages)		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to complex numbers Standard form (a + bi) Arithmetic: addition, subtraction, multiplication	Book1: Chapter 1
Week 2	Division of complex numbers Polar (trigonometric) form De Moivre's Theorem nth roots of complex number	Book1: Chapter 1
Week 3	Sequences and sigma notation Arithmetic sequences and series Applications	Book1: Chapter 6 Assignment 1
Week 4	Geometric sequences and series Finite and infinite geometric series	Book1: Chapter 6 Quiz 1
Week 5	Counting & Probability Permutations and combinations Basic probability concepts	Book1: Chapter 7
Week 6	Binomial Theorem Expansion using binomial theorem	Book1: Chapter 7

	General term Applications in algebra and probability	
Week 7	Limits (Foundations) Concept and notation of limits Finding limits using graphs and tables Direct substitution	Thomas Jr. et al.: Chapter 2 Assignment 2
Week 8	Limits (Techniques & Continuity) Rationalization techniques One-sided limits Continuity and types of discontinuities	Thomas Jr. et al.: Chapter 2 Quiz 2
Week 9	Rate of Change Average vs instantaneous rate of change Tangent lines and slopes Real-life applications	Thomas Jr. et al.: Chapter 3
Week 10	Definition of derivative Basic differentiation rules Power rule	Thomas Jr. et al.: Chapter 3
Week 11	Product and quotient rules Chain rule Derivatives of trigonometric functions	Thomas Jr. et al.: Chapter 3 Assignment 3
Week 12	Derivatives of exponential and logarithmic functions Derivatives of inverse trigonometric functions Applications of derivatives	Thomas Jr. et al.: Chapter 3 Quiz 3
Week 13	Applications of Derivatives Increasing and decreasing functions Concavity and inflection points Relative extrema Absolute maxima and minima	Thomas Jr. et al.: Chapter 4
Week 14	Area problem and intuition of integrals Indefinite integrals Basic integration rules Substitution method	Thomas Jr. et al.: Chapter 5 Quiz 4
Week 15	Definite integrals Area under a curve	Thomas Jr. et al.: Chapter 5 Assignment 4
Week 16	Definition of area as a limit (using sigma notation)	Thomas Jr. et al.: Chapter 5

COMPUTER SCIENCE CORE: 48 (41,7)

Sr.	Code	Course Title	Prerequisite	Cr. Hrs.
1.	CSC-110	Programming Fundamentals		3 (3,0)
2.	CSC-110-L	Programming Fundamentals Lab		1 (0,1)
3.	CSC-111	Object Oriented Programming	CSC-110	3 (3,0)
4.	CSC-111-L	Object Oriented Programming Lab	CSC-110	1 (0,1)
5.	CSC-112	Digital Logic Design		2 (2,0)
6.	CSC-112-L	Digital Logic Design Lab		1 (0,1)
7.	CSC-210	Data Structures	CSC-111	3 (3,0)
8.	CSC-210-L	Data Structures Lab	CSC-111	1 (0,1)
9.	CSC-211	Database Systems		3 (3,0)
10.	CSC-211-L	Database Systems Lab		1 (0,1)
11.	CSC-212	Design & Analysis of Algorithms	CSC-210	3 (3,0)
12.	CSC-213	Operating Systems		3 (3,0)
13.	CSC-213-L	Operating Systems Lab		1 (0,1)
14.	CSC-214	Software Engineering		3 (3,0)
15.	CSC-215	Computer Organization & Architecture (Assembly language included)	CSC-112	3 (2,1)
16.	CSC-310	Computer Networks		3 (3,0)
17.	CSC-311	Artificial Intelligence		3 (3,0)
18.	CSC-312	Theory of Automata		3 (3,0)
19.	CSC-313	Information Security		3 (3,0)
20.	CSC-314	Cloud Computing		3 (3,0)

Course Title	**Programming Fundamentals**		
Course Code	**CSC-110**		
Credit Hours	3 (3,0)		
Category	Computer Science Core		
Prerequisite	None		
Co-Requisite	None		
Follow Up	CSC-111 Object Oriented Programming		
Course Introduction	This course provides fundamental concepts of programming to freshmen. The course is a prerequisite to many other courses; therefore, students are strongly advised to cover all the content and try to achieve CLOs to the maximum possible level.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Demonstrate proficiency in writing, debugging, and executing basic programs using programming languages such as C++ or Python.	C3 (Apply)	1,2,4,5
	CLO2: Explain core programming concepts including variables, control structures, functions, and data types.	C2 (Understand)	1,2
	CLO3: Develop simple algorithms to solve computational problems.	C6 (Create)	1,2,3,4
	CLO4: Apply best practices in coding to produce efficient and readable programs.	C3 (Apply)	1,2,4,5,8,9
	CLO5: Analyze program outputs and troubleshoot common errors effectively.	C4 (Analyze)	1,2,3,5,10
Text Book(s)	1. Tony Gaddis, Starting out with C++, 10th Edition, Pearson, 2024.		
Reference Material	1. Object Oriented Programming in C++ by Robert Lafore 2. Problem Solving and Program Design in C++, 7th Edition by Jeri R. Hanly & Elliot B. Koffman		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to Problem Solving: Problem solving steps, algorithms, pseudo-code, flowcharts, overview of Von-Neumann Architecture	Gaddis: Chapter 1 Handouts
Week 2	Overview of the structured program theorem (sequence, selection and iteration)	Gaddis: Chapter 1 Handouts Quiz 1
Week 3	Introduction to C++: Structure of a C++ program, compilation process (compiler & linker), development environment, simple program (printing text), Variables and Data Types: Concept of variables, data types (int, float, char, bool)	Gaddis: Chapter 2 Assignment 1
Week 4	Input/output using cin/cout, arithmetic expressions, assignment statements, Operators and Expressions: Arithmetic operators, operator precedence, integer division, type casting, introduction to memory concepts	Gaddis: Chapter 3 Quiz 2
Week 5	Decision Making I: Relational and equality operators, if and if-else statements, nested if	Gaddis: Chapter 4
Week 6	Decision Making II & Loops: Logical operators, while loop, structured program development	Gaddis: Chapter 4-5 Quiz 3
Week 7	Iteration Control: for, do-while loops, break and continue, introduction to switch statement	Gaddis: Chapter 5 Assignment 2

Week 8	Functions I: Modular programming, function definition and prototypes, passing arguments by value	Gaddis: Chapter 6 Quiz 4
Week 9	Functions II: Passing by reference, scope rules, recursion, recursion vs iteration, random number generation	Gaddis: Chapter 6
Week 10	Arrays I: One-dimensional arrays, initialization, processing arrays, passing arrays to functions	Gaddis: Chapter 7 Quiz 5
Week 11	Arrays II: Searching and sorting arrays, multidimensional arrays, character arrays (strings basics)	Gaddis: Chapter 8 Assignment 3
Week 12	Strings and Structures: C++ strings, string handling functions, structures, accessing members, typedef, unions	Gaddis: Chapter 9-10 Quiz 6
Week 13	Pointers I: Pointer concepts, declaration and initialization, pointer operators, pointers and arrays	Gaddis: Chapter 9-10
Week 14	Pointers II & Dynamic Memory: Pointer arithmetic, array of pointers, function pointers (intro), dynamic memory allocation (new/delete)	Gaddis: Chapter 9-10 Quiz 7
Week 15	File Handling & Preprocessor: File streams (read/write), sequential files, basics of random access, preprocessor directives (#include, #define, macros, assertions)	Gaddis: Chapter 12 Assignment 4
Week 16	Advanced Topics & Review: Bitwise operators, enumerations, command-line/variable arguments, program termination (exit), course review	Gaddis: Chapter 11

Course Title	**Programming Fundamentals Lab**		
Course Code	**CSC-110-L**		
Credit Hours	1 (0,1)		
Category	Computer Science Core		
Prerequisite	None		
Co-Requisite	None		
Follow Up	CSC-111 Object Oriented Programming		
Course Introduction	This course provides fundamental concepts of programming to freshmen. The course is a prerequisite to many other courses; therefore, students are strongly advised to cover all the content and try to achieve CLOs to the maximum possible level.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Demonstrate proficiency in writing, debugging, and executing basic programs using programming languages such as C++ or Python.	C3 (Apply)	1,2,4,5
	CLO2: Explain core programming concepts, including variables, control structures, functions, and data types.	C2 (Understand)	1,2
	CLO3: Develop simple algorithms to solve computational problems.	C6 (Create)	1,2,3,4
	CLO4: Apply best practices in coding to produce efficient and readable programs.	C3 (Apply)	1,2,4,5,8,9
	CLO5: Analyze program outputs and troubleshoot common errors effectively.	C4 (Analyze)	1,2,3,5,10
Text Book(s)	1. Tony Gaddis, Starting out with C++, 10th Edition, Pearson, 2024.		
Reference Material	1. Object Oriented Programming in C++ by Robert Lafore 2. Problem Solving and Program Design in C++, 7th Edition by Jeri R. Hanly & Elliot B. Koffman		

Weekly Lesson Plan

Week No.	Lab Content	Assignments/Readings
Week 1	Lab Orientation: Introduction to problem solving, algorithms, pseudo-code, flowcharts; writing first pseudo-code	Gaddis: Chapter 1 Handouts Lab Tasks
Week 2	Practice implementation of sequence, selection and iteration	Gaddis: Chapter 1 Handouts Lab Tasks
Week 3	overview of C++ IDE (Dev C++, Visual Studio etc.), First C++ Programs: Structure of C++ program, compiling & running	Gaddis: Chapter 2 Lab Tasks
Week 4	Printing text, simple input/output using cin/cout; Variables & Data Types: Declaring variables, basic data types, arithmetic expressions, assignment statements; Operators & Expressions	Gaddis: Chapter 2-3 Lab Tasks
Week 5	Decision Making I: Relational and logical operators; if and if-else programs (marks grading, max/min, etc.)	Gaddis: Chapter 4 Lab Tasks
Week 6	Decision Making II: Nested if, introduction to while loop; solving repetitive problems	Gaddis: Chapter 4-5 Lab Tasks
Week 7	Iterative Structures: for, do-while, switch, break, continue; menu-driven programs	Gaddis: Chapter 5 Lab Tasks

Week 8	Functions I: Function definition, prototypes, calling functions, passing arguments by value; basic modular programs	Gaddis: Chapter 6 Lab Tasks
Week 9	Functions II: Pass by reference, scope rules, recursion (factorial, Fibonacci), random number generation	Gaddis: Chapter 6 Lab Tasks
Week 10	Arrays I: One-dimensional arrays, input/output, summation, searching elements	Gaddis: Chapter 7 Lab Tasks
Week 11	Arrays II: Sorting (bubble/selection), multidimensional arrays, matrix operations	Gaddis: Chapter 8 Lab Tasks
Week 12	Strings & Structures: Character arrays vs string, string handling, defining structures, accessing members	Gaddis: Chapter 9-10 Lab Tasks
Week 13	Pointers I: Pointer basics, initialization, pointer operators, pointers with arrays	Gaddis: Chapter 9-10 Lab Tasks
Week 14	Pointers II & Dynamic Memory: Pointer arithmetic, dynamic memory (new/delete), passing by reference using pointers	Gaddis: Chapter 9-10 Lab Tasks
Week 15	File Handling & Preprocessor: File streams (ifstream, ofstream), reading/writing files, intro to macros (#define)	Gaddis: Chapter 12 Lab Tasks
Week 16	Advanced Topics & Final Practice: Bitwise operators, enums, debugging practice, mini problem-solving session / lab test	Gaddis: Chapter 11 Lab Tasks

Course Title	**Object Oriented Programming**		
Course Code	**CSC-111**		
Credit Hours	3 (3,0)		
Category	Computer Science Core		
Prerequisite	CSC-110 Programming Fundamentals		
Co-Requisite	None		
Follow Up	CSC-210 Data Structures		
Course Introduction	The course aims to focus on object-oriented concepts, analysis and software development. The basic concept of OOP is covered in this course.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Understand and apply the principles of object-oriented programming, including encapsulation, inheritance, and polymorphism.	C3 (Apply)	1,2
	CLO2: Design and implement classes and objects to model real-world entities.	C3 (Apply)	3,4,5
	CLO 3: Write reusable and modular code using OOP concepts.	C3 (Apply)	2,4,5
	CLO4: Analyze the advantages of OOP over procedural programming paradigms.	C4 (Analyze)	2,3
	CLO5: Develop small-scale applications utilizing OOP concepts in languages like Java or C++.	C6 (Develop)	4,5,6
Text Book(s)	P. Deitel, H. Deitel, C++ How To Program, 10th Edition, Pearson.		
Reference Material	1. Robert Lafore, Object Oriented Programming in C++, 3rd Edition. 2. Tony Gaddis, Starting Out with C++ from Control Structures to Objects, 9th Edition, Pearson, 2018. 3. Problem Solving and Program Design in C++, 7th Edition by Jeri R. Hanly & Elliot B. Koffman		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to Object-oriented Design, History and Advantages of Object-oriented Design. Overview of C++ Concepts and Structured Programming: • Pointer/Alias, Arrays, Dynamic Memory Allocation • C/C++ struct keyword: Data Driven Programming • Passing/Returning structures by reference/value Introduction to Object Oriented Concepts and Terminologies using real world examples	Lafore: Chapter 1
Week 2	Introduction to Classes, Objects and Member Functions: Define the keyword 'class' Data Abstraction through Encapsulation; Declare a class and define data members Access specifiers/modifiers: private and public only Information Hiding through Access Specifiers First Member function; Setter/Mutator and Getter/Accessor methods; How the function knows which object invoked it? (answered later in course)	Lafore: Chapter 6
Week 3	Two Special Method: Constructor & Destructor; Constructor with No argument (Default Constructor);	Lafore: Chapter 6 Assignment 1

	Overloaded Constructors, Constructor with Default arguments Constructor Delegation Member initializer list Dynamic Memories and Significance of Destructors 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	
Week 4	Passing objects to functions by value and potential problem; Default Member-wise copy (Assignment & Initialization), Problem of Member-wise copy & its solution: Copy Constructor (concepts of Shallow Copy vs. Deep Copy), Calling sequence of Constructor & Destructor: when objects are passed by value/reference Preventing changes in data members from a method: 'const' keword; Constant data members Static functions Static data members Static member functions (absence of this pointer) constant and static objects Calling sequence of Constructor & Destructor for constant and static objects Constant member functions and their need for Constant objects Can a Static member function be declared 'const'?	Lafore: Chapter 6 Quiz 2
Week 5	Composition and Aggregation • Object as data member (Composition) • Object reference as a data member (Composition/Aggregation) Cascading calls with and without this pointer. Nameless objects Array of objects;	Lafore: Chapter 6 Assignment 2
Week 6	Operator Overloading; Binary Operator receiving Instance of class as 1st Operand Overloading Unary Operator: as member, as non-member Unusual Operators: ++, --, Dynamic Memory Management while overloading assignment operator '='	Lafore: Chapter 8
Week 7	Overloading logical operators Overloading [ ] (set & get, both versions), type-cast and function call operator - Parentheses ()	Lafore: Chapter 8 Quiz 3
Week 8	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); Operators as Members vs. Non-Members: Overloading I/O stream binary operators as friend functions: • stream insertion operator <<, • stream extraction operator >>	Lafore: Chapter 8 Assignment 3

	Overloading Binary operators receiving only 2nd operand in the parameter list, NOT receiving instance of class as 1st operand Conversion between Types, Explicit Constructor, Type-Cast Constructor and Conversion Operators	
Week 9	Inheritance is-A Relationship: Public Inheritance Protected data member protected, private inheritance 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;	Lafore: Chapter 9
Week 10	Polymorphism: Overriding base-class members in derived class Virtual functions and Dynamic binding Concept of v-Pointer and v-Table	Lafore: Chapter 10, 11
Week 11	Abstract classes Pure virtual functions and abstract class; Defining a pure virtual destructor Polymorphism under the Hood Detail discussion on v-table structure, virtual constructor, object cloning and Concept of Downcasting, keywords 'dynamic_cast' and 'nullptr'	Lafore: Chapter 11 Quiz 4
Week 12	Multiple Inheritance Diamond Problem (sprouting from multiple inheritance), Virtual Inheritance A review of OO relationships terminologies: Association (knows-a) unidirectional and bidirectional Aggregation (has-a), Composition (owns-a), Generalization (is-a) . UML Notation	Lafore: Chapter 9, 15 Assignment 4
Week 13	C++ Streams, Members and Manipulators of Streams; File Handling using Streams Access Techniques: Sequential, Direct, and Random Access Files; Input/Output of Object from/to File Opening Modes (binary/ text mode) Processing Methods (binary / formatted)	Lafore: Chapter 12
Week 14	Templates; Function Templates, Class Templates Template Specialization Function Template Specializations (compared with function overloading) Class Template Specializations Specialized Methods of Class Template Suitability of Templates for Container Classes	Lafore: Chapter 14 Quiz 5
Week 15	Partial Template Specialization Partial Class Template specialization (not available for function templates) Friendship and Inheritance with Templates Standard Template Library, Vectors, foreach loop	Lafore: Chapter 14, 15 Assignment 5

Week 16	Exception Handling: Error vs. Exception Evolution of Exception Handling: from exit, abort, assert to new keywords: try, catch, throw Propagation of Exception and its advantage Unhandled Exception; Overall Review of Course for Final Exam	Lafore: Chapter 14 Quiz 6

Course Title	**Object Oriented Programming Lab**		
Course Code	**CSC-111-L**		
Credit Hours	1 (0,1)		
Category	Computer Science Core		
Prerequisite	CSC-110 Programming Fundamentals		
Co-Requisite	None		
Follow Up	CSC-210 Data Structures		
Course Introduction	The course aims to focus on object-oriented concepts, analysis and software development. The basic concept of OOP is covered in this course.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Understand and apply the principles of object-oriented programming, including encapsulation, inheritance, and polymorphism.	C3 (Apply)	1,2
	CLO2: Design and implement classes and objects to model real-world entities.	C3 (Apply)	3,4,5
	CLO 3: Write reusable and modular code using OOP concepts.	C3 (Apply)	2,4,5
	CLO4: Analyze the advantages of OOP over procedural programming paradigms.	C4 (Analyze)	2,3
	CLO5: Develop small-scale applications utilizing OOP concepts in languages like Java or C++.	C6 (Develop)	4,5,6
	CLO6: Develop proficiency in debugging and refactoring object-oriented code to enhance readability, efficiency, and maintainability.	C6 (Create)	2,3,4, 5,6,7,9
Text Book(s)	P. Deitel, H. Deitel, C++ How To Program, 10th Edition, Pearson.		
Reference Material	1. Robert Lafore, Object Oriented Programming in C++, 3rd Edition. 2. Tony Gaddis, Starting Out with C++ from Control Structures to Objects, 9th Edition, Pearson, 2018. 3. Problem Solving and Program Design in C++, 7th Edition by Jeri R. Hanly & Elliot B. Koffman		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Revision and Hands-on practice exercises on: • Pointer/Alias, • Arrays, • Dynamic Memory Allocation • Modular Programming using Structs and Functions	Lafore: Chapter 1 Lab Tasks
Week 2	Getting familiar with keyword 'class'. Defining a Class Defining Data members Writing wrapper methods and basic functionalities for the class	Lafore: Chapter 6 Lab Tasks
Week 3	Object instantiation using different constructor Cleaning up resources and destroying objects using destructor.	Lafore: Chapter 6 Lab Tasks
Week 4	Focus the use of const and static keywords with data member, member functions and class instances (objects). Targets the use/purpose of class level information.	Lafore: Chapter 6 Lab Tasks

Week 5	Object manipulation and transition in case of aggregation/composition and arrays of objects.	Lafore: Chapter 6 Lab Tasks
Week 6	More exercises on composition and aggregation for clarity Two way relationship of objects using Association Basic operator overloading focusing on binary operators	Lafore: Chapter 6 Lab Tasks
Week 7	Operatorover loading -arithmatic opeators -Logical operators - increment ++, decrement -- operators Special Operators [], (),	Lafore: Chapter 8 Lab Tasks
Week 8	Friend functions in perspective of operator overloading: • Overloading I/O stream binary operators as friend functions: stream insertion operator <<, stream extraction operator >> • Overloading Binary operators receiving only 2nd operand in the parameter list, NOT receiving instance of class as 1st operand	Lafore: Chapter 8 Lab Tasks
Week 9	Inheritance: • Public Inheritance • Declaring 'protected' data members in base class to allow access to derived classes • protected, private inheritance • Multilevel Inheritance	Lafore: Chapter 9 Lab Tasks
Week 10	Learning the basic advantage of using polymorphism in Inheritance hierarchy	Lafore: Chapter 10 Lab Tasks
Week 11	Another lab to practice Polymorphism with focus on pure virtual functions and abstract classes.	Lafore: Chapter 10-11 Lab Tasks
Week 12	Practicing to map different OOP concepts to real world problems: Association (knows-a) unidirectional and bidirectional, Aggregation (has-a), Composition (owns-a), Generalization (is-a) Polymorphism	Lafore: Chapter 11 Lab Tasks
Week 13	Command Line Arguments, File Stream (Formatted I/O)	Lafore: Chapter 13 Lab Tasks
Week 14	Function Templates and Class Templates, Full Specialization of Function Templates and Class Templates	Lafore: Chapter 14 Lab Tasks
Week 15	Class templates Partial Specializations, C++ Standard Templat Library (STL)	Lafore: Chapter 14 Lab Tasks
Week 16	Exception Handling using C++ 'exception' Class	Lafore: Chapter 14-15 Lab Tasks

Course Title	**Digital Logic Design**		
Course Code	**CSC-112**		
Credit Hours	3 (3,0)		
Category	Computer Science Core		
Prerequisite	None		
Co-Requisite	None		
Follow Up	CSC-215 Computer Organization & Architecture (Assembly language included)		
Course Introduction	The course introduces the concept of digital logic, gates and the digital circuits. Further, it focuses on the design and analysis combinational and sequential circuits. It also serves to familiarize the student with the logic design of basic computer hardware components.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Draw and interpret digital logic diagrams, including combinational and sequential circuits.	C2 (Understand)	1, 2, 3
	CLO 2: Design basic digital components such as multiplexers, flip-flops, and encoders.	C6 (Create)	1,3,4,5
	CLO 3: Analyze the behaviour of digital systems using truth tables and Boolean algebra.	C4 (Analyze)	3
	CLO 4: Implement simple digital circuits using logic gates.	C3 (Apply)	4, 5
	CLO 5: Understand the fundamentals of digital system design and their applications.	C2 (Understand)	1,2,8,10
Text Book(s)	1. M. Morris Mano, Digital Logic and Computer Design, 1st Edition, Pearson, 1979, ISBN:0132145103.		
Reference Material	1. Thomas L. Floyd, Digital Fundamentals, 10th Edition, Prentice Hall, 2008, ISBN: 0132359235. 2. Fundamental of Digital Logic with Verilog Design, Stephen Brown, 2/e		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to Digital Systems, Number Systems (base conversions, data encodings , radix and diminished radix complements and applying in subtraction)	Mano: Chapter 1
Week 2	Introduction to Boolean Algebra, Basic theorems and properties of Boolean Algebra,	Mano: Chapter 2 Quiz 1
Week 3	Boolean Functions. Canonical Forms (Representation of Function in Sum of Minterms or Product of Maxterms,), Standard Forms.	Mano: Chapter 2 Assignment 1
Week 4	Simplification of Boolean function using Boolean Algebra	Mano: Chapter 2
Week 5	Logic Gates, NAND and NOR Implementation,	Mano: Chapter 2 Quiz 2
Week 6	Simplification of Boolean function using Karnaugh Map,	Mano: Chapter 3 Assignment 2
Week 7	Don't care Conditions in K Maps	Mano: Chapter 3
Week 8	The Tabulation Method	Mano: Chapter 3 Quiz 3
Week 9	Introduction to Combinational Logic, Design of Adders, Design of Subtractors, Code Convertors, Analysis Procedure of Combinational Circuits	Mano: Chapter 4 Assignment 3

Week 10	Binary Parallel Adders, Decimal Adders, Magnitude Comparator, Decoders and its applications, Multiplexers, Demultiplexers, Encoders, ROM, Programmable Logic Array (PLA),	Mano: Chapter 5
Week 11	Introduction to Sequential Circuits, Basic Flip Flop, Clocked RS Flip Flop, Clocked D Flip Flop, Clocked JK Flip Flop, Clocked T Flip Flop,	Mano: Chapter 6 Quiz 4
Week 12	Analysis of Clocked Sequential Circuits, State Reduction and Assignment,	Mano: Chapter 6 Assignment 4
Week 13	Introduction to Registers, Shift Registers, Ripple Counters, Synchronous Counters , Timing Sequences	Mano: Chapter 7
Week 14	Memory Unit, Random Access Memory	Mano: Chapter 7 Quiz 5
Week 15	Introduction Programmable Logic Devices	Mano: Chapter 7 Assignment 5
Week 16	Verilog HDL/VHDL, MultiSim Familiarization (or anyother SPICE Simulation Tool) SPICE: Simulation Program with Integrated Circuit Emphasis	Mano: Chapter 4 - 6

Course Title	**Digital Logic Design Lab**		
Course Code	**CSC-112-L**		
Credit Hours	1 (0,1)		
Category	Computer Science Core		
Prerequisite	None		
Co-Requisite	None		
Follow Up	CSC-215 Computer Organization & Architecture (Assembly language included)		
Course Introduction	The course introduces the concept of digital logic, gates and the digital circuits. Further, it focuses on the design and analysis combinational and sequential circuits. It also serves to familiarize the student with the logic design of basic computer hardware components.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Draw and interpret digital logic diagrams, including combinational and sequential circuits.	C2 (Understand)	1,2,3
	CLO 2: Design basic digital components such as multiplexers, flip-flops, and encoders.	C6 (Create)	1
	CLO 3: Analyze the behaviour of digital systems using truth tables and Boolean algebra.	C4 (Analyze)	3
	CLO 4: Implement simple digital circuits using logic gates.	C3 (Apply)	1
	CLO 5: Understand the fundamentals of digital system design and their applications.	C2 (Understand)	1,2,8,10
Text Book(s)	M. Morris Mano, Digital Logic and Computer Design, 1st Edition, Pearson, 1979, ISBN:0132145103.		
Reference Material	1. Thomas L. Floyd, Digital Fundamentals, 10th Edition, Prentice Hall, 2008, ISBN: 0132359235. 2. Fundamental of Digital Logic with Verilog Design, Stephen Brown, 2/e		

Weekly Lesson Plan

Week No.	Course Content	Readings/Learning Activities
Week 1	Familiarization with the Digital Electronic Trainer.	Mano: Chapter 1 Lab Tasks
Week 2	Implementation of logic gates using ICs, AND Gate Operation OR Gate Operation Not Gate Operation	Mano: Chapter 2 Lab Tasks
Week 3	NAND Gate Operation NOR Gate Operation Conversion of NAND and NOR gates	Mano: Chapter 2 Lab Tasks
Week 4	Construction of XOR Gate from NAND&NOR Gate Construction of XNOR Gate from NAND and NOR Gate Construction of any circuit using NAND-only logic	Mano: Chapter 2 Lab Tasks
Week 5	Illustration of basic properties and theorems Boolean algebra using circuit design, Implementation of given Boolean functions	Mano: Chapter 2 Lab Tasks
Week 6	Using K-Maps for Function simplification and implementation	Mano: Chapter 3 Lab Tasks
Week 7	Half Adder, Full Adder	Mano: Chapter 3 Lab Tasks
Week 8	Half Subtractor,	Mano: Chapter 3

	Full Subtractor	Lab Tasks
Week 9	Design of 7-Segment Display	Mano: Chapter 4 Lab Tasks
Week 10	Decoder BCD To 7-Segment Display	Mano: Chapter 5 Lab Tasks
Week 11	Multiplexer, Demultiplexer	Mano: Chapter 6 Lab Tasks
Week 12	Implementation of Boolean function(s) using Decoder and Multiplexer	Mano: Chapter 6 Lab Tasks
Week 13	Magnitude Comparator	Mano: Chapter 7 Lab Tasks
Week 14	D Latch and Flip-Flop Operation Latching BCD Data for Displaying On 7- Segment Display	Mano: Chapter 7 Lab Tasks
Week 15	Shift Register Operation JK Flip-Flop Operation	Mano: Chapter 7 Lab Tasks
Week 16	Random Access Memories	Mano: Chapter 4 – 6 Lab Tasks

Course Title	**Data Structures**		
Course Code	**CSC-210**		
Credit Hours	3 (3,0)		
Category	Computer Science Core		
Prerequisite	CSC-111 Object Oriented Programming		
Co-Requisite	None		
Follow Up	CSC-212 Design & Analysis of Algorithms		
Course Introduction	The course is designed to teach students structures and schemes, which allow them to write programs to efficiently manipulate, store, and retrieve data. Students are exposed to the concepts of time and space complexity of computer programs.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Implement and analyze various data structures such as arrays, linked lists, stacks, queues, trees, and graphs.	C3 (Solve) C4 (Analyze)	2,3,4
	CLO2: Select appropriate data structures to optimize algorithm performance.	C3 (Apply)	2,4
	CLO3: Demonstrate proficiency in traversing and manipulating data structures.	C3 (Solve)	2,4
	CLO4: Evaluate the efficiency of algorithms based on data structure choices.	C2 (Understand)	2,3
	CLO5: Solve complex problems using suitable data structures and algorithms.	C3 (Solve)	2,3,4
Text Book(s)	1. Data Structures and Algorithms in C++ by Adam Drozdek		
Reference Material	1. Classic Data Structures by Debasis Samanta 2. Data Structures and Algorithm Analysis in C++ by Mark Allen Weiss 3. C++ Plus Data Structures by Nell Dale		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to Data Structures and Algorithms: Understanding Their Role in Computer Science, Defining Algorithms, Properties of Algorithms, and the Relationship Between Data Structures and Algorithms, Revision OOP Concepts, Abstract Data Types, and Encapsulation	Drozdek: Chapter 1
Week 2	Inheritance, Pointers and Arrays, Polymorphism, Iterators and Algorithms, Computational and Asymptotic Complexity	Drozdek: Chapter 2 Quiz 1
Week 3	Big-O Notation, Properties of Big-O Notation, Finding Asymptotic Complexity: Examples, The Best, Average, and Worst Cases.	Drozdek: Chapter 2 Assignment 1
Week 4	Linear List with Array Implementation. Singly Linked List, Insertion, Deletion, Traversal	Drozdek: Chapter 3 Quiz 2
Week 5	Circular List, Doubly Linked List Stacks, Standard Functions with their Implementation	Drozdek: Chapter 3
Week 6	Stack Problems. Brackets Validation, In-Order to Post-Order Conversion, Maze Finding. Queue and Circular Queue	Drozdek: Chapter 4 Quiz 3
Week 7	Queue Problems. Process/ Job scheduling Recursive Definition and Processes: Understanding Direct Recursion and Learning Recursive Trace; Implementing Linear Search and Binary Search.	Drozdek: Chapter 4 Assignment 2

Week 8	Recursion Continued: Implementing Merge Sort and Quick Sort. Backtracking	Drozdek: Chapter 5 Quiz 4
Week 9	Binary Tree, Root Node, Height, Parent/ Child Relation, Tree Traversals. Binary Tree Array Implementation, Recursive Implementation of Operations	Drozdek: Chapter 6
Week 10	Binary Tree Linked Implementation, Traversals, Height Binary Search Tree, Insertion, Deletion, Search, Count	Drozdek: Chapter 6 Quiz 5
Week 11	Height Balanced Trees – AVL Trees AVL Tree Implementation	Drozdek: Chapter 6 Assignment 3
Week 12	Heaps, Insertion, Deletion, Heap Sort. Heap as Priority Queue	Drozdek: Chapter 6 Quiz 6
Week 13	Hashing, Hash Functions, Hash Advanced Functions, Collision Resolution, Open Hashing	Drozdek: Chapter 10
Week 14	Graph, Depth First Search, Breadth First Search. Graph Implementations, Adjacency Matrix, Adjacency List	Drozdek: Chapter 8 Quiz 7
Week 15	Cycle Detection, Shortest Path, Minimal Spanning Tree	Drozdek: Chapter 8 Assignment 4
Week 16	Topological Sort, Graph Coloring, Revision, and Summary	Drozdek: Chapter 8

Course Title	**Data Structures Lab**		
Course Code	**CSC-210-L**		
Credit Hours	1 (0,3)		
Category	Computer Science Core		
Prerequisite	CSC-111 Object Oriented Programming		
Co-Requisite	None		
Follow Up	None		
Course Introduction	The course is designed to teach students structures and schemes, which allow them to write programs to efficiently manipulate, store, and retrieve data. Students are exposed to the concepts of time and space complexity of computer programs.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Implement and analyze various data structures such as arrays, linked lists, stacks, queues, trees, and graphs.	C3 (Solve) C4 (Analyze)	2,3,4
	CLO2: Select appropriate data structures to optimize algorithm performance.	C3 (Apply)	2,4
	CLO3: Demonstrate proficiency in traversing and manipulating data structures.	C3 (Solve)	2,4
	CLO4: Evaluate the efficiency of algorithms based on data structure choices.	C2 (Understand)	2,3
	CLO5: Solve complex problems using suitable data structures and algorithms.	C3 (Solve)	2,3,4
Text Book(s)	1. Data Structures and Algorithms in C++ by Adam Drozdek		
Reference Material	1. Classic Data Structures by Debasis Samanta 2. Data Structures and Algorithm Analysis in C++ by Mark Allen Weiss 3. C++ Plus Data Structures by Nell Dale		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Implementation of Set & Matrix ADT	Drozdek: Chapter 1 Lab Tasks
Week 2	Bubble Sort, Selection Sort, and Insertion Sort with Time Calculation	Drozdek: Chapter 9 Lab Tasks
Week 3	Implementation of an Array-Based Linear List	Mark Allen: Chapter 3 Lab Tasks
Week 4	Singly Linked List Functions – add at start, add at end, remove from start, remove from end, search, display, reverse, etc.	Drozdek: Chapter 3 Lab Tasks
Week 5	Doubly Linked List Functions – sort, count, collapse Implementation of a doubly-linked circular list	Drozdek: Chapter 3 Lab Tasks
Week 6	Check Palindrome, Parenthesis Matching, Array Reverse, Pre-order to Post-Order Conversion	Drozdek: Chapter 4 Lab Tasks
Week 7	Processor Queue Simulation, Ticket Purchase in Queue One at a Time.	Drozdek: Chapter 4 Lab Tasks
Week 8	Recursive Algorithms – Factorial – Power – Patterns	Drozdek: Chapter 5 Lab Tasks
Week 9	Maze Finding – Array Target Sum	Drozdek: Chapter 5 Lab Tasks
Week 10	Binary Tree Array and Linked Implementation	Drozdek: Chapter 6 Lab Tasks

Week 11	Binary Search Tree insertion, deletion, search, and count	Drozdek: Chapter 6 Lab Tasks
Week 12	AVL Tree Implementation	Drozdek: Chapter 6 Lab Tasks
Week 13	Priority Queue for Patient Queue with Emergency Patient	Drozdek: Chapter 6 Lab Tasks
Week 14	Storing and Retrieval of Key Words of C++	Drozdek: Chapter 10 Lab Tasks
Week 15	Basic Graph Implementation, Shortest Path	Drozdek: Chapter 8 Lab Tasks
Week 16	Spanning Tree & miscellaneous graph problems, like graph coloring	Drozdek: Chapter 8 Lab Tasks

Course Title	**Database Systems**		
Course Code	**CSC-211**		
Credit Hours	3 (3,0)		
Category	Computer Science Core		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	The course aims to introduce basic database concepts, different data models, data storage and retrieval techniques and database design techniques. The course primarily focuses on relational data model and DBMS concepts.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Design and normalize relational database schemas based on user requirements.	C4 (Analyze)	3,4
	CLO2: Write SQL queries for data retrieval, insertion, update, and deletion.	C3 (Apply)	2,5
	CLO3: Explain the concepts of database transactions, concurrency, and recovery.	C2 (Understand)	2
	CLO4: Implement basic database management tasks using popular database management systems.	C3 (Apply)	4,5
	CLO5: Analyze the role of databases in information systems and their security considerations.	C4 (Analyze)	8,9
Text Book/ Reference Material	1. Carlos Coronel, Steven Morris, Database Systems: Design, Implementation & Management, 13th Edition, Cengage Learning, 2017. ISBN-10: 1337627909. 2. Introduction to Oracle 9i: SQL By Oracle Press 3. Jeffrey A. Hoffer, Ramesh Venkataraman, Heikki Topi, Modern Database Management, 12th Edition, Pearson, 2015. ISBN-10: 0133544613. 4. Thomas Connolly, Carolyn Begg, Database Systems: A Practical Approach to Design, Implementation and Management, 6th Edition, Pearson, 2015. ISBN-10: 1292061189. 5. Ramez Elmasri, Shamkant B. Navathe, Fundamentals of Database Systems, 7th Edition, Pearson, 2016. ISBN-10: 1292097612.		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	File Systems and Databases: Introducing the Database; Why Database Design is important, A practical approach to Database Design. Historical roots of the database; Files and File system, A File systems Critique; File System Data Management, Structural and Data dependence, Field Definitions and Naming Conventions, Data Redundancy. Database Systems: Database system Environment, Types of Database Management Systems, DBMS Functions, Managing the Database System, Database Design and Modeling.	Coronel: Chapter 1
Week 2	Database Models: Hierarchical Model, Network Model, Relational Model, Entity Relationship Data Model, Object-Oriented Model, The Evolution of Data Models; Database Models and the Internet. Introduction to RDBMS: A logical view of Data; Entities and Attributes, Tables and their Characteristics, Keys.	Coronel: Chapter 1,2,3 Assignment 1
Week 3	SQL: Writing Basic SQL Statements; SELECT Statement, Arithmetic Expressions, Operator Precedence, Null Value, Column Alias, Concatenation Operator, Display table Structure. Integrity Rules: Entity Integrity, Referential Integrity Relational Database Operators: Data	SQL By Oracle Press: Chapter 1 Coronel: Chapter 3 Quiz 1

	Dictionary and System Catalog.	
Week 4	SQL: Restricting and Sorting Data; WHERE Clause, Comparison operators, Logical operators, ORDER BY clause, ASC order, Desc Order. Entity Relationship (E-R) Modeling: Basic Modeling Concepts; Data Models; Degrees of Data Abstraction; Conceptual Model, Internal Model, External Model, Physical Model.	SQL By Oracle Press: Chapter 2 Coronel: Chapter 4
Week 5	SQL: Single Row Functions; Character Functions; Case Conversion, Character manipulation, Date Functions. Entity Relationship (E-R) Model: Entities, Attributes, Relationships, Connectivity and Cardinality, Relationship Strength (Existence Dependency), Relationship Participation, Relationship Strength and weak entities, Relationship Degree.	SQL By Oracle Press: Chapter 3 Coronel: Chapter 4 Assignment 2
Week 6	SQL: Single Row Functions; Data type conversion Functions, General Functions. Entity Relationship (E-R) Model: Composite Entities, Entity Super types and subtypes, Comparison of E-R Modeling Symbols.	SQL By Oracle Press: Chapter 3 Coronel: Chapter 4
Week 7	SQL: Displaying Data from Multiple Tables, Equi-Join, Non-Equi Join, Outer Join, Self-Join. Developing an E-R Diagram: Challenge of Database Design, Conflicting Goals. The System Development Life Cycle (SDLC), The Database Life Cycle (DBLC), Database Design Strategies.	SQL By Oracle Press: Chapter 4 Coronel: Chapter 4 Quiz 2
Week 8	SQL: Aggregating Data Using Group Functions; Group Functions, GROUP BY clause, HAVING clause. Transform E-R Diagram into Database Structure: General Rules Governing Relationships Among Tables.	SQL By Oracle Press: Chapter 5 Coronel: Chapter 4
Week 9	SQL: Sub queries, Types of Sub queries, single row sub query, Group functions in sub query, HAVING clause in sub query, Multiple row sub query, multiple row comparison operator	SQL By Oracle Press: Chapter 6 Coronel: Chapter 5
Week 10	Normalization of Database Tables: Need for Normalization, Conversion to First Normal Form, Conversion to Second Normal Form.	SQL By Oracle Press: Chapter 6 Coronel: Chapter 5 Assignment 3
Week 11	SQL: Sub queries; Multiple Column Sub queries, pair wise comparison, Non-pair wise comparison, Null Value in a sub query, Sub query in From Clause. Normalization: Conversion to Third Normal Form, Boyce-Codd Normal Form (BCNF).	SQL By Oracle Press: Chapter 7 Coronel: Chapter 5 Quiz 3
Week 12	SQL: Manipulating Data, Creating and Managing Tables; the INSERT statement, UPDATE statement, DELETE statement, CREATE table statement, ALTER Table statement, DROP table statement, RENAME statement, TRUNCATE table statement. Normalization and Database Design: Higher Level Normal Forms, De-Normalization	SQL By Oracle Press: Chapter 9-10 Coronel: Chapter 5
Week 13	SQL: Defining Constraints, Column Level and Table Level, NOT NULL Constraint, UNIQUE Key Constraint, PRIMARY Key Constraint, FOREIGN Key Constraint, CHECK Constraint. Transaction Management and Concurrency Control: What is a Transaction; Evaluating Transaction Results, Transaction Properties, Transaction Management with SQL, Transaction Log, Transaction Types.	SQL By Oracle Press: Chapter 11 Coronel: Chapter 10 Assignment 4
Week 14	SQL: Creating views; Simple view, Complex views, CREATE View statement, Rules for DML operations, WITH CHECK OPTION clause, Denying DML operations, DROP view Statement. Concurrency Control with locking Methods: Lost Updates, Un-committed Data, Inconsistent Retrievals, Dirty Data, Fuzzy Read, Scheduler. Lock Granularity, Lock Types, Two-Phase Locking to Ensure Serializability, Deadlocks.	SQL By Oracle Press: Chapter 12 Coronel: Chapter 10 Quiz 4
Week 15	Small organization Database Design, Development and Connectivity Concurrency control with Time Stamping Methods: Concurrency control with optimistic Methods. Database Recovery Management; Transaction Recovery, Architectural Considerations, Recovery information.	Coronel: Chapter 10

Week 16	SQL: Producing Readable Outputs with SQL* PLUS, other Database objects. DDBMS: Evolution, Advantages & Disadvantages. Distributed processing and distributed databases.What is a DDBMS?, Components.	SQL By Oracle Press: Chapter 17 Coronel: Chapter 11

Course Title	**Database Systems Lab**		
Course Code	**CSC-211-L**		
Credit Hours	1 (0,1)		
Category	Computer Science Core		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	The course aims to introduce the Structured Query Language (SQL). It covers the set of commands related to Data Retrieval, Data Definition Language (DDL), Data Manipulation Language (DML), and Data Control Language (DCL). It will be followed up by the procedural flavor of SQL (PL/SQL).		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Design and normalize relational database schemas based on user requirements.	C4 (Analyze)	3,4
	CLO2: Write SQL queries for data retrieval, insertion, update, and deletion.	C3 (Apply)	2,5
	CLO3: Explain the concepts of database transactions, concurrency, and recovery.	C2 (Understand)	2
	CLO4: Implement basic database management tasks using popular database management systems.	C3 (Apply)	4,5
	CLO5: Analyze the role of databases in information systems and their security considerations.	C4 (Analyze)	8,9
Text Book/ Reference Material	1. Introduction to Oracle 9i: SQL By Oracle Press 2. Michael McLaughlin, Oracle Database 11g PL/SQL Programming, 1st Edition, McGrawHill Education, 2008, ISBN: 0071494456. 3. 3. Jason Price, Oracle Database 11g SQL, McGraw Hill. ISBN: 0071498508.		

Weekly Lesson Plan		
Week No.	**Course Content**	**Assignments/Readings**
Week 1	Installation of Oracle/MS SQL Server, Introduction to SQL	Lab Assignment 1 SQL By Oracle Press: Chapter 1
Week 2	SQL: Writing Basic SQL Statements; SELECT Statement, Arithmetic Expressions, Operator Precedence, Null Value, Column Alias, Concatenation Operator, Display table Structure	Lab Assignment 2 SQL By Oracle Press: Chapter 1
Week 3	SQL: Restricting and Sorting Data; WHERE Clause, Comparison operators, Logical operators, ORDER BY clause, ASC order, Desc Order	Lab Assignment 3 SQL By Oracle Press: Chapter 2
Week 4	SQL: Single Row Functions; Character Functions; Case Conversion, Character Manipulation, Date Functions	Lab Assignment 4 SQL By Oracle Press: Chapter 3
Week 5	SQL: Single Row Functions; Data type conversion Functions, General Functions	Lab Assignment 5 SQL By Oracle Press: Chapter 3

Week 6	SQL: Displaying Data from Multiple Tables, Equi-Join, Non-Equi Join, Outer Join, Self Join	Lab Assignment 6 SQL By Oracle Press: Chapter 4
Week 7	SQL: Aggregating Data Using Group Functions; Group Functions, GROUP BY clause, HAVING clause	Lab Assignment 7 SQL By Oracle Press: Chapter 5
Week 8	SQL: Sub queries, Types of Sub queries, single row sub query, Group functions in sub query, HAVING clause in sub query, Multiple row sub query, multiple row comparison operator	Grand Quiz 1 Lab Assignment 8 SQL By Oracle Press: Chapter 6
Week 9	SQL: Sub queries; Multiple Column Sub queries, pair wise comparison, Non-pair wise comparison, Null Value in a sub query, Sub query in From Clause.	Lab Assignment 9 SQL By Oracle Press: Chapter. 7
Week 10	SQL: Manipulating Data, Creating and Managing Tables; the INSERT statement, UPDATE statement, DELETE statement, CREATE table statement, ALTER Table statement, DROP table statement, RENAME statement, TRUNCATE table statement	Lab Assignment 10 SQL By Oracle Press: Chapter. 9&10
Week 11	SQL: Defining Constraints, Column Level and Table Level, NOT NULL Constraint, UNIQUE Key Constraint, PRIMARY Key Constraint, FOREIGN Key Constraint, CHECK Constraint.	Lab Assignment 11 SQL By Oracle Press: Chapter. 11
Week 12	SQL: Creating views; Simple view, Complex views, CREATE View statement, Rules for DML operations, WITH CHECK OPTION clause, Denying DML operations, DROP view Statement	Lab Assignment 12 SQL By Oracle Press: Chapter. 12
Week 13	SQL: Producing Readable Outputs with SQL* PLUS, other Database objects.	Term Project Viva Lab Assignment 13 SQL By Oracle Press: Chapter. 16
Week 14	Introduction to PL/SQL, Sections of a PL/SQL block, Variable declaration and initialization.	Lab Assignment 14 SQL By Oracle Press: Chapter. 17
Week 15	PL/SQL: Selection, Repetition, Exception, Handling, Cursors	Lab Assignment 15 SQL By Oracle Press: Chapter. 17
Week 16	PL/SQL: Stored Procedure, Functions and Trigger	Grand Quiz 2 Lab Assignment 16 SQL By Oracle Press: Chapter. 17

Course Title	**Design & Analysis of Algorithms**		
Course Code	**CSC-212**		
Credit Hours	3 (3,0)		
Category	Computer Science Core		
Prerequisite	CSC-210 Data Structures		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course aims to cultivate students' thinking abilities and proficiency in algorithm design and analysis. It teaches the process of algorithm design, emphasizing efficiency in terms of time and space, providing strategies for optimization, guiding the selection of appropriate data structures, fostering a systematic problem-solving approach, and last but not least, an awareness that not all problems are solvable with the introduction to P vs NP problems.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Design efficient algorithms for common computational problems.	C6 (Create)	2,4
	CLO2: Analyze algorithm complexity using Big-O notation.	C4 (Analyze)	2,3
	CLO3: Solve problems involving recursion, divide-and-conquer, and dynamic programming.	C3 (Apply)	3,4
	CLO4: Compare different algorithmic approaches to problem-solving.	C5 (Evaluate)	3
	CLO5: Demonstrate correctness and optimality of algorithms.	C3 (Evaluate)	3,4
Text Book(s)	1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, Introduction to Algorithms, 4th Edition, The MIT Press, 2022, ISBN-10: 026204630X, ISBN-13: 978-0262046305		
Reference Material	1. Algorithm Design, (1st edition, 2013/2014), Jon Kleinberg, Eva Tardos 2. Algorithms, (4th edition, 2011), Robert Sedgewick, Kevin Wayne		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to Course, Review of Data Structure and some Discrete Mathematics concepts, Role of Algorithms in Computing, Insertion Sort and its line-by-line analysis, and its correctness	Chapter 1, Chapter 2
Week 2	Asymptotic Notations: Big O, Little O, Big Omega, Little Omega, etc. Introduction to the Divide and Conquer Approach, Merge Sort, and its analysis.	Chapter 3 Quiz 1
Week 3	Recurrences and methods to solve recurrence equations: The Substitution Method, Recursion Tree Method, The Master Method, Description of Quick sort, Performance of Quick sort	Chapter 4, Chapter 7 Assignment 1
Week 4	Heaps, Maintaining a Heap Property, Building a Heap, Heap Sort Algorithm, Priority Queues	Chapter 6 Quiz 2
Week 5	Lower Bound on comparison-based sorting, Counting Sort, Radix Sort, Bucket Sort	Chapter 8
Week 6	Minimum and maximum, Selection in expected linear time, Selection in worst-case linear time, Introduction to Dynamic Programming, Matrix Chain Multiplication	Chapter 9, Chapter 14 Quiz 3
Week 7	0-1 Knapsack problem, Coin-change problem, Assembly-line Scheduling	Chapter 14 Assignment 2
Week 8	Longest common subsequence, Optimal binary search trees	Chapter 14

		Quiz 4
Week 9	Greedy Algorithm, An activity-selection problem, Elements of Greedy Strategy, Knapsack Problem, Variant of Knapsack problem, Huffman Code	Chapter 15
Week 10	Graph Review, Representation of Graph	Chapter 20 Quiz 5
Week 11	Breadth First Search (BFS), Applications of BFS, Depth First Search (DFS), Applications of DFS	Chapter 20 Assignment 3
Week 12	Topological Sort, Strongly Connected Components, Minimum Spanning Tree, Kruskal Algorithm	Chapter 20, Chapter 21 Quiz 6
Week 13	Prim's Algorithm, Shortest Path Algorithm, Bellman-Ford Algorithm	Chapter 21,
Week 14	Single Source Shortest Path in a directed acyclic graph, Dijkstra's Algorithm, Floyd Warshall Algorithm	Chapter 22, Chapter 23 Quiz 7
Week 15	String Matching with Finite Automata, Knuth-Morris-Pratt Algorithm	Chapter 32 Assignment 4
Week 16	NP-completeness and the classes P and NP, Decision problems vs. optimization problems, NP-completeness and reducibility, NP-complete problems	Chapter 34

Course Title	**Operating Systems**		
Course Code	**CSC-213**		
Credit Hours	3 (3,0)		
Category	Computer Science Core		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	To help students gain a general understanding of the principles and concepts governing the functions of operating systems and acquaint students with the layered approach that makes design, implementation, and operation of the complex OS possible.		
	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Explain the functions and services provided by operating systems.	C2 (Understand)	1,2
	CLO2: Manage processes, threads, and synchronization mechanisms.	C2 (Understand)	2
	CLO3: Analyze memory management techniques and file systems.	C4 (Analyze)	3
	CLO 4: Implement basic scheduling algorithms.	C3 (Demonstrate)	3,4
	CLO 5: Evaluate operating system performance and security features.	C5 (Evaluate)	4
Text Book(s)	1. A. Silberschatz, P. B. Galvin, G. Gagne, Operating Systems Concepts, 9th Edition, Wiley, 2012, ISBN: 1118063333.		
Reference Material	1. Andrew S. Tanenbaum, Herbert Bos, Modern Operating Systems, 4th Edition, Pearson, 2014, ISBN: 013359162X. 2. William Stallings, Operating Systems: Internals and Design Principles, 9th Edition, Pearson, 2017, ISBN: 0134670957.		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction of the course; Overview of pre-requisite concepts/knowledge, What is Operating System?, System Goals, History of Operating Systems. Mainframe Systems, Batch Systems, Multi-programmed Systems. Time-Sharing Systems, Desktop Systems, Multiprocessor Systems, Distributed Systems, Client-Server Systems, Real-Time Systems, Handheld Systems, Computer-System Operation	(Evolution of OS) Silberschatz: Chapter 1 Assignment 1
Week 2	OS structures, Introduction to VMWare, Hardware Protection, Dual-Mode Operation, I/O Protection, Memory Protection, CPU Protection, System Components, Operating System Services Interrupt, Trap and Signals, System Calls, OS Kernel, File Management in UNIX, PPFDT, Reading and writing using System calls in UNIX, Redirection operators in UNIX.	Silberschatz: Chapter 2
Week 3	Process Concept, Process Control Block, Process States, Process Scheduling. Process creation and termination, fork(), wait() and exec() system calls.	Silberschatz: Chapter 3 Quiz 1
Week 4	Cooperating Processes, Producer Consumer Problem, Interprocess Communication (Direct and Indirect). UNIX IPC tools	Silberschatz: Chapter 3

Week 5	Introduction to threads, multi-threading and hyper-threading, advantages of threads, user-level and kernel-level threads, Threading models, Programming using Pthread library	Silberschatz: Chapter 4 Quiz 2
Week 6	CPU scheduling and scheduling criteria, FCFS, SJF, and SRTF. Priority Scheduling, Round Robin, Virtual Round Robin, Multi-level Queue Scheduling, and Multi-level Feedback Queue Scheduling, Evaluation of Scheduling algorithms.	Silberschatz: Chapter 6 Assignment 2
Week 7	Introduction to synchronization, Concurrency Control, Race Condition, Critical Section Problem, Characteristics of CS problem solution. Software-based solution to CS problem, Peterson solution, Bakery algorithm,	Silberschatz: Chapter 5 Assignment 3
Week 8	H/W based solution; disabling of interrupts, TSL and swap instruction.	Silberschatz: Chapter 5 Quiz 3
Week 9	Thread synchronization using mutex, Introduction to semaphore, binary and counting semaphores, Achieving mutual exclusion using semaphores, Achieving serialization using semaphores.	Silberschatz: Chapter 5
Week 10	Solution to Standard Synchronization problems using semaphores, Producer Consumer, Dining Philosopher, Reader Writer, Sleeping Barber, Smokers problem. Limitations of Semaphores	Silberschatz: Chapter 5
Week 11	Introduction to Deadlocks, Four necessary and sufficient conditions for Deadlocks, Resource allocation graph, Deadlock handling methods, Deadlock prevention, Deadlock avoidance, Bankers and Safety Algorithm, Deadlock detection and recovery	Silberschatz: Chapter 7 Quiz 4 Assignment 4
Week 12	Memory Management, address Binding and linking, Logical vs Physical Addresses, Dynamic Loading, Dynamic Linking & Shared Libraries, Overlays, Swapping, Introduction to Contiguous Memory Allocation. MFT and MVT, Placement algorithms, Internal and External fragmentation, Buddy partitioning scheme.	Silberschatz: Chapter 8
Week 13	Introduction to paging, Page Table, Address translation in paging, 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	Silberschatz: Chapter 8 Quiz 5
Week 14	Introduction to paged segmentation, address translation in a paged segmentation. Address translation in Intel 80386 (Real and protected mode) Virtual Memory, Background, Demand Paging, Performance of Demand Paging, Page Replacement algorithms (FIFO, Optimal)	Silberschatz: Chapter 9 Assignment 5
Week 15	Other page replacement algorithms (, LRU, LFU, MFU, Buffering), Copy on Write protocol and vfork() system call, Allocation Of Frames, Thrashing, Resident Set Management, Working Set Model, Page fault frequency, memory mapped files.	Silberschatz: Chapter 9
Week 16	Free Space Management (bit vector, linked list, grouping, counting), Disk quotas, File Protection (ACLs and Capabilities), Protection in UNIX File caching, File locking, File system reliability, Encryption, Disk Scheduling Algorithms, Disk partitioning, Disk formatting, Performance and reliability using RAID	Silberschatz: Chapter 10

Course Title	**Operating Systems Lab**		
Course Code	**CSC-213-L**		
Credit Hours	1 (0,1)		
Category	Computer Science Core		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	The Operating Systems lab course provides a hands-on environment where students bridge the gap between theoretical resource management and practical system-level programming. Designed to complement the core lecture series, the course transitions from mastering the Linux command-line interface and automation through shell scripting to implementing complex kernel-level logic using C/C++. Students engage in deep-dive exercises involving process synchronization, multithreading with Pthreads, and the simulation of CPU scheduling and memory management algorithms. By directly interacting with system calls and managing concurrent executions, students develop the technical proficiency required to analyze, debug, and optimize the underlying software that powers modern computing hardware.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Apply Linux/Unix command-line tools and shell scripting to automate administrative tasks and manage the system environment.	C3 (Apply)	3,4
	CLO2: Implement system-level programs in C/C++ using system calls for process management, multithreading, and inter-process communication.	C3 (Apply)	3,4
	CLO3: Analyze the logic and performance of various CPU scheduling and page replacement algorithms through software simulation.	C4 (Analyze)	3,4
	CLO4: Construct robust solutions for classical synchronization and deadlock problems using mutex locks, semaphores, and avoidance logic.	C6 (Construct)	5.6
	CLO5: Evaluate the efficiency of memory management and file system operations based on empirical data collected during lab experiments.	C5 (Evaluate)	5,6
Text Book(s)	1. Operating Systems Concepts, 10th edition, Abraham Silberschatz, Peter Baer Galvin, Greg Gagne; Wiley.		
Reference Material	1. Lab Manual handbook and programming exercises provided by the course instructor.		

Weekly Lesson Plan		
Week No.	**Course Content**	**Assignments/Readings**
Week 1	Explore the Linux terminal, directory navigation (cd, ls, pwd), and file manipulation (cp, mv, rm). Understand user permissions and the Linux file hierarchy.	Appendix A: Linux/UNIX Programming Environment (pp. A-1 – A-19) Chapter 1-2 Lab Tasks
Week 2	Introduction to Bash scripting: variables, positional parameters, and standard I/O redirection. Write scripts to automate basic system tasks.	Appendix A: Linux/UNIX Programming

		Environment (pp. A-1 – A-19) Chapter 2 Lab Tasks
Week 3	Advanced scripting using if-else logic, case statements, and loops (for, while). Practice parsing system logs or managing user accounts.	Appendix A: Linux/UNIX Programming Environment (pp. A-1 – A-19) Chapter 2 Lab Tasks
Week 4	Use the fork() system call to create processes. Analyze the parent-child relationship and practice using getpid() and getppid().	Chapter 3 Lab Tasks
Week 5	Implementation of exec() family calls, wait(), and exit(). Create scenarios to observe Zombie and Orphan process behavior.	Chapter 3 Lab Tasks
Week 6	Hands-on with Shared Memory (shmget, shmat) and Message Passing (Pipes). Implement a basic data exchange between two processes.	Chapter 3 Lab Tasks
Week 7	Use the Pthreads library in C to create and join threads. Compare the overhead of thread creation versus process creation.	Chapter 4 Lab Tasks
Week 8	Practical Evaluation: A timed lab exam covering Shell scripting, fork() operations, and basic IPC implementations.	Lab Midterm Exam
Week 9	Code the simulation for First-Come, First-Served (FCFS) and Shortest Job First (SJF) algorithms. Calculate average waiting and turnaround times.	Chapter 5 Lab Tasks
Week 10	Implement Round-Robin (RR) and Priority Scheduling. Observe the effect of different time quantums on process throughput.	Chapter 5 Lab Tasks
Week 11	Simulate a "Race Condition." Use Mutex Locks to protect a critical section and ensure data consistency in a multi-threaded environment.	Chapter 6 Lab Tasks
Week 12	Solve the Producer-Consumer Problem using Semaphores. Implement synchronization to manage a shared bounded buffer.	Chapter 6-7 Lab Tasks
Week 13	Implement the Banker's Algorithm. Write a program that takes resource requests and determines if the system remains in a "Safe State".	Chapter 8 Lab Tasks
Week 14	Simulate Paging logic: Implement a program to map logical addresses to physical addresses using a mock page table.	Chapter 9 Lab Tasks
Week 15	Implement FIFO, LRU, and Optimal page replacement algorithms. Compare the number of page faults generated by each under specific reference strings.	Chapter 10 Lab Tasks
Week 16	Final Evaluation: Comprehensive lab exam, such as a custom Shell or a complex synchronization simulation.	Final Lab Exam

Course Title	**Software Engineering**		
Course Code	**CSC-214**		
Credit Hours	3 (3,0)		
Category	Computer Science Core		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Apply software development life cycle (SDLC) models to manage projects effectively	C3 (Apply)	4,10
	CLO2: Develop software requirements specifications and design documents	C6 (Create)	3,4
	CLO3: Implement and test software applications using best practices	C3 (Apply)	4,5
	CLO4: Analyze and manage software project risks and quality	C4 (Analyze)	3,4
	CLO5: Collaborate effectively in team-based software projects	C3 (Apply)	6,7
Text Book(s)	1. Ian Sommerville, Software Engineering, 10th Edition, Pearson, 2015, ISBN-13: 978-0133943030. 2. Software Engineering Software Engineering: A Practitioner's Approach 6th edition, ISBN-13: 978-0071238403		
Reference Material	1. Carlo Ghezzi, Mehdi Jazayeri, Dino Mandrioli, Fundamentals of Software Engineering, 2nd Edition, Pearson, 2002, ISBN-13: 978-0133056990.		

Weekly Lesson Plan		
Week No.	**Course Content**	**Assignments/Readings**
Week 1	The Scope of Software Engineering: Motivation and need for software engineering, Definition of Software Engineering, Introduction to software engineering vocabulary, Software Process Models: Introduction, Linear Sequential Model, Prototyping Model	Sommerville: 1.1, 2.3 Assignment 1
Week 2	RAD Model, Evolutionary software process models, Incremental Model, Spiral Model, Extreme Programming, Introduction to Requirements Engineering	Sommerville: 2.3 Quiz 1
Week 3	Introduction to Structured Analysis and Design, Structured Analysis: Functional Modeling (DFD, Process Specification, Mini Specification)	Sommerville: 6.4
Week 4	DFD Case Study, Structured Analysis: Behavioral Modeling (STD, Control Specification)	Sommerville: 7.2 Quiz 2
Week 5	Structured Design: Basic Concepts, Data Design and Introduction to Architectural Design, Architectural Design and mapping of requirements to architectural design	Sommerville: 8.3, 9.1, 9.3, 9.6 Assignment 2
Week 6	Component Level Design, User Interface Design, Object Oriented Analysis & Design Basics	Sommerville: 10.5, 11.1, 11.2 Quiz 3
Week 7	Introduction to UML, UML Diagrams, Use Case Modelling	Pressman: 1.1, 1.2, 1.3, 1.4, 1.5 Assignment 3

Week 8	System level use case Diagram, Use case modelling in Rational Rose, Activity Diagram	Pressman: 6.1, 6.2, 6.3, 6.4, 6.5, 6.6, 6.7,6.8, 6.9, 6.12, 6.13, 6.16,6.17 Quiz 4, Assignment 4
Week 9	Introduction of Interaction Diagrams, System Sequence Diagram, Domain Model: Identifying business classes, Domain Model Associations, Domain Model Attributes	Pressman: 8.2, 8.3, 8.5, 8.6, 9.2, 9.3, 9.4, 9.5, 9.6, 9.7, 9.8, 9.9, 9.10
Week 10	Implementation of Sequence Diagram, activity diagram and Domain model in Rational Rose, Use case Operational Contracts	Pressman: 8.2, 8.3, 8.5, 8.6, 9.2, 9.3, 9.4, 9.5, 9.6, 9.7, 9.8, 9.9, 9.10 Assignment 5
Week 11	Interaction Diagram: Sequence diagrams, Collaboration Diagrams, Use case relationships, Analysis level use case Diagram	Pressman: 8.2, 8.3, 8.5, 8.6, 9.2, 9.3, 9.4, 9.5, 9.6, 9.7, 9.8, 9.9, 9.10 Quiz 5, Assignment 6
Week 12	Implementation of Sequence, Collaboration and analysis level use case diagrams in Rational Rose, Introduction to design patterns, GRASP, Information Expert, Creator	Pressman: 8.2, 8.3, 8.5, 8.6, 9.2, 9.3, 9.4, 9.5, 9.6, 9.7, 9.8, 9.9, 9.10 Assignment 7
Week 13	GRASP: Cohesion, Coupling, Controller, Introduction to Design Class Diagram, Mapping data model to Domain model, Implementation of Design Class Diagram in Rational Rose	Pressman: 16.7,16.8, 16.9,16.10 Pressman: 19.1, 19.4, 19.5, 19.6 Quiz 6
Week 14	Other Patterns: Assigning responsibilities, Polymorphism, Indirection, Protected variation	Handouts Assignment 8
Week 15	Mapping Design to Code	Pressman: 20.1, 20.2, 20.3, 20.4, 20.5, 20.7, 20.9, 20.11 Quiz 7
Week 16	Software Testing Fundamentals	Handouts Quiz 8

Course Title	**Computer Organization & Architecture**		
Course Code	**CSC-215**		
Credit Hours	3 (2,1)		
Category	Computer Science Core		
Prerequisite	CSC-112 Digital Logic Design		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course introduces the fundamental concepts of computer organization and architecture, focusing on how hardware components interact to execute programs. It covers CPU organization, instruction set architecture, memory systems, and input/output mechanisms. Basic assembly language programming is included to help students understand low-level execution. The course emphasizes conceptual understanding rather than detailed hardware design.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Describe the structure and function of computer components (CPU, memory, I/O)	C2 (Understand)	1,2
	CLO2: Write and understand basic assembly language programs	C3 (Apply)	2,5
	CLO3: Analyze how hardware components interact during program execution	C4 (Analyze)	3
	CLO4: Explain concepts of instruction set architecture and microarchitecture	C2 (Understand)	2
	CL05: Optimize programs considering hardware limitations	C5 (Evaluate)	3,4
Text Book(s)	1. M. Morris Mano, Computer System Architecture, 3rd Edition, Pearson 2. William Stallings, Computer Organization and Architecture, 10th Edition		
Reference Material	1. Patterson & Hennessy, Computer Organization and Design 2. Carl Hamacher, Computer Organization		

Weekly Lesson Plan		
Week No.	**Course Content (CLO)**	**Assignments/Readings**
Week 1	Introduction to Computer Organization and Architecture, basic functional units (CPU, Memory, Input/Output), stored program concept	Mano: Chapter 1, Stallings: Chapter 1
Week 2	Number systems (binary, octal, hexadecimal), signed number representation (1's & 2's complement), basic arithmetic operations	Mano: Chapter 2, Hamacher: Chapter 2 Quiz 1
Week 3	Review of digital logic: registers, register transfer language (RTL), introduction to ALU operations (conceptual)	Mano: Chapter 4, Stallings: Chapter 12
Week 4	Basic computer organization: bus system, register organization, instruction flow inside CPU	Mano: Chapter 5, Patterson & Hennessy: Chapter 4
Week 5	Instruction Set Architecture (ISA): instruction types, instruction formats, operation codes, operands	Stallings: Chapter 13, Patterson & Hennessy: Chapter 2 Quiz 2
Week 6	nstruction cycle: Fetch–Decode–Execute cycle, role of control unit (hardwired vs microprogrammed – overview)	Mano: Chapter 5, Stallings: Chapter 15
Week 7	Addressing modes: Immediate, Direct, Indirect, Indexed; examples and comparison	Mano: Chapter 5, Hamacher: Chapter 4 Assignment 1

Week 8	Introduction to Assembly Language: basic syntax, instruction types, registers usage	Mano: Chapter 6, Patterson & Hennessy: Chapter 2 Quiz 3
Week 9	Assembly Programming: arithmetic operations, loop structures, simple program development	Mano: Chapter 6, Instructor Notes
Week 10	Assembly Programming: conditional branching, flags, simple decision-making programs	Mano: Chapter 6, Patterson & Hennessy: Chapter 2
Week 11	Memory hierarchy: cache, main memory, concept of locality, performance considerations (no heavy numericals)	Stallings: Chapter 4–5, Patterson & Hennessy: Chapter 5 Quiz 4
Week 12	Input/Output organization: programmed I/O, interrupts, basic DMA concept	Stallings: Chapter 7, Mano: Chapter 7
Week 13	RISC vs CISC architectures: characteristics, advantages, real-world relevance	Patterson & Hennessy: Chapter 1, Stallings: Chapter 18 Assignment 2
Week 14	Introduction to pipelining: concept, stages, basic idea of speedup and hazards (conceptual only)	Patterson & Hennessy: Chapter 4, Stallings: Chapter 14 Quiz 5
Week 15	Overview of MIPS architecture: instruction types (R, I, J), simple examples	Patterson & Hennessy: Chapter 2–3
Week 16	Course review and integration of concepts	Review All Relevant Chapters

Course Title	**Computer Networks**		
Course Code	**CSC-310**		
Credit Hours	3 (3,0)		
Category	Computer Science Core		
Prerequisite	None		
Co-Requisite	None		
Follow-up	None		
Course Learning Outcomes (CLOs)	At the end of the lab, the students will be able to:	**BT**	**PLO**
	CLO1: Describe the key terminologies and technologies of computer networks	C2 (Describe)	1
	CLO2: Explain the services and functions provided by each layer in the Internet protocol stack	C2 (Explain)	1
	CLO3: Identify various internetworking devices and protocols and their functions in a network	C4 (Identify)	1,2
	CLO4: Analyze the working and performance of key technologies, algorithms, and protocols	C4 (Analyze)	3
	CLO5: Build Computer Network on various Topologies.	C6 (Build)	4,5
Course Description	Introduction: 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, subnetting, CIDR, routing protocols, transport layer protocols, ports and sockets, connection establishment, flow and congestion control, application layer protocols, latest trends in computer networks. The lab contents are divided into two parts. After completion of the first part, the students will be able to understand Computer Networks basics, network types, layered communication models and protocols. The contents have been organized in such a way as to form the base for learning the concepts to be taught in the next part. The assimilation of the networking concepts will enable the students to apply them for solving practical problems. After completion of this part, the students will be familiar with networking, routing, switching, and Setting up of networks from scratch, major protocols involved in communication and their configurations. Students will also be made aware of the state of the art areas in case they would like to pursue this course in future.		
Text Book(s)	1. James F. Kurose and Keith W. Ross, Computer Networking: A Top-Down Approach Featuring the Internet, 6th Edition, Pearson, 2012, ISBN: 0132856204. 2. T. Lammle, CCNA Cisco Certified Network Associate Deluxe Study Guide, 6th Edition, Sybex, 2011, ISBN: 978-0-470-90108-3.		
Reference Material	1. Andrew S. Tanenbaum, David J. Wetherall, Computer Networks, 5th Edition, Prentice Hall, 2010, ISBN: 9332518742. 2. William Stallings, Data and Computer Communications, 10th Edition, Pearson, 2013, ISBN: 0133506487. 3. Behrouz A. Forouzan, Data Communication and Computer Networks, 5th Edition, McGraw- Hill, 2012, ISBN: 0073376221. 4. R. Perlman, Interconnections: Bridges, Routers, Switches, and Internetworking Protocols, 2nd Edition, Addison-Wesley, 1999, ISBN: 0201634481.		

Weekly Lesson Plan		
Week No.	**Course Content**	**Assignments/Readings**
Week 1	What Is the Internet? The Network Edge, The Network Core	Kurose: Chapter 1

Week 2	Delay, Loss, and Throughput in Packet-Switched Networks.	Kurose: Chapter 1
Week 3	Protocol Layers and Their Service Models.	Kurose: Chapter 1
Week 4	Principles of Network Application. The Web and HTTP.	Kurose: Chapter 2
Week 5	Electronic Mail in the Internet. DNS—The Internet's Directory Service.	Kurose: Chapter 2
Week 6	Peer-to-Peer File Distribution, Video Streaming and Content DistributionNetworks.	Kurose: Chapter 2
Week 7	Introduction and Transport-Layer Services, Multiplexing and De-multiplexing, Connectionless Transport: UDP	Kurose: Chapter 3
Week 8	Principles of Reliable Data Transfer	Kurose: Chapter 3
Week 9	Connection-Oriented Transport: TCP. Principles of Congestion Control, TCP Congestion Control.	Kurose: Chapter 3
Week 10	Overview of Network Layer. What's Inside a Router?	Kurose: Chapter 4
Week 11	The Internet Protocol (IP): IPv4, Addressing, IPv6, and More. Generalized Forwarding and SDN.	Kurose: Chapter 4
Week 12	Introduction, Routing Algorithms. Intra-AS Routing in the Internet: OSPF.	Kurose: Chapter 5
Week 13	Routing Among the ISPs: BGP. The SDN Control Plane.	Kurose: Chapter 5
Week 14	ICMP: The Internet Control Message Protocol, Network Management andSNMP, NETCONF/YANG.	Kurose: Chapter 5
Week 15	Introduction to the Link Layer, Error-Detection and Correction Techniques	Kurose: Chapter 6
Week 16	Multiple Access Links and Protocols	Kurose: Chapter 6

Course Title	**Artificial Intelligence**		
Course Code	**CSC-311**		
Credit Hours	3 (3,0)		
Category	Computer Science Core		
Prerequisites	CSC-210 Data Structures, GE-164 Probability & Statistics		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course introduces the fundamental principles and techniques of Artificial Intelligence. It covers classical symbolic AI, heuristic search, and logic-based reasoning while providing a modern pivot toward connectionist approaches, specifically Deep Learning and Generative AI. Students will explore how to build intelligent agents that perceive, reason, and act within various environments.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Describe core AI concepts including search algorithms, knowledge representation, and reasoning.	C2 (Understand)	1,2
	CLO2: Implement basic AI algorithms for problem-solving and decision-making.	C3 (Apply)	3,4
	CLO3: Analyze the applications of AI in real-world scenarios.	C4 (Analyze)	3,4
	CLO4: Discuss ethical considerations and limitations of AI systems.	C2 (Understand)	8,9
	CLO5: Develop simple AI models using appropriate tools and frameworks.	C6 (Create)	5,6,7
Text Book(s)	1. Artificial Intelligence: A Modern Approach by Stuart Russell & Peter Norvig, (4th Edition, Global Education, 2020); ISBN-13: 978-0134610993.		
Reference Material	1. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (3rd Ed. 2022) by Aurélien Géron. 2. Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play (2nd Ed. 2023) by David Foster 3. Deep learning, Ian Goodfellow, Yoshua Bengio, Aaron Courville, the MIT press, (2018). 4. Artificial Intelligence- Structures & Strategies for Complex Problem Solving by George F. Luger, 5th Edition, Pearson Education (2008); ISBN-13: 978- 0321545893.		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction & Intelligent Agents: Defining AI, Turing Test, Rationality, PEAS Framework, and Agent Architectures. AIMA Ref: Chapters 1 & 2.	Activity: Discussion on Weak vs. Strong AI and modern chatbot success criteria. Russell & Norvig: Chapters 1 & 2
Week 2	Uninformed Search Strategies: State Space Representation, BFS, DFS, Depth-Limited Search, and Iterative Deepening. AIMA Ref: Chapter 3.	Russell & Norvig: Chapter 3
Week 3	Informed Search & Heuristics: Greedy Best-First Search, A* Algorithm, and Heuristic	Russell & Norvig: Chapter 3

	properties (Admissibility & Consistency). AIMA Ref: Chapter 3 (3.5–3.6).	Quiz 1
Week 4	Local Search Algorithms and Optimization Problems: Hill-climbing search, Simulated annealing and Local beam search. AIMA Ref: Chapter 4.	Russell & Norvig: Chapter 4 Quiz 2
Week 5	Adversarial Search: Game Theory, Optimal Decisions in Games, Minimax Algorithm, and Alpha-Beta Pruning, Forward and backward chaining. AIMA Ref: Chapter 5 (5.1–5.3), Ch 7 (7.5).	Russell & Norvig: Chapters 5 & 7 Assignment 1
Week 6	Probabilistic Reasoning: Quantifying Uncertainty, Bayes' Rule and its Use, and Naive Bayes Models. AIMA Ref: Chapters 12.	Russell & Norvig: Chapter 12
Week 7	Introduction to Machine Learning: Regression Forms of learning, learning Decision Trees, and Linear Regression. AIMA Ref: Chapter 19 & Handouts.	Russell & Norvig: Chapter 19 & Handouts Quiz 3
Week 8	Introduction to Machine Learning: Classification Classification & Clustering. AIMA Ref: Chapter 19 & Handouts.	Russell & Norvig: Chapter 19 & Handouts
Week 9	The Connectionist Shift: Perceptrons, Multi-layer Perceptrons (MLP), Backpropagation, and the transition to Transformers. AIMA Ref: Chapter 22 & Handouts.	Project Activity: GenAI Term Project Assigned. Russell & Norvig: Chapter 22 & Handouts
Week 10	Generative AI & Large Language Models (LLMs): Transformer Architecture (Self-Attention), Tokenization, and LLM training (Pre-training vs. Fine-tuning). AIMA Ref: Chapter 25 (Deep Learning for NLP). Foster Ref: Chapter 1 (Generative Modeling).	Russell & Norvig: Chapter 25 (Deep Learning for NLP). Foster Ref 2: Chapter 1 (Generative Modelling) Assignment 2
Week 11	RAG (Retrieval-Augmented Generation): What is RAG, Context Windows, Embedding Models. Foster Ref: Chapter 10 (Advanced Transformers).	Foster: Chapter 10 (Advanced Transformers) Quiz 4
Week 12	RAG 2: Vector Databases (FAISS/Chroma). RAG implementation Foster Ref: Chapter 10 (Advanced Transformers).	Foster: Chapter 10 (Advanced Transformers)
Week 13	Prompt Engineering & Reasoning: Zero-shot/Few-shot Learning, Chain-of-Thought Prompting, and Hallucinations. Foster Ref: Chapter 9 (Transformers).	Foster: Chapter 9 (Transformers) Quiz 5
Week 14	AI Ethics, Safety & Alignment Bias in Generative Models, Deepfakes, AI Privacy, and Professional Ethics. AIMA Ref: Chapter 28 (Philosophy & Ethics).	Project Prototype Russell & Norvig: Chapter 28
Week 15	Future of AI: Agentic Workflows (AutoGPT), AGI, and course recap.	

Week 16	Final Project Demos & Presentations	Project Demo & Presentations

Course Title	**Theory of Automata**		
Course Code	**CSC-312**		
Credit Hours	3 (3,0)		
Category	Computer Science Core		
Prerequisite	None		
Co-Requisite	None		
Follow Up	CSE-326 Compiler Construction		
Course Introduction	This course introduces the fundamental concepts of automata theory and formal languages. It covers finite automata, regular expressions, grammars, pushdown automata, and Turing machines, focusing on their design, analysis, equivalence, and computational limitations. Applications in compiler design and language processing are also explored.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Explain the concepts of finite automata, regular expressions, and formal languages.	C2 (Understand)	1,2
	CLO2: Design automata to recognize specific languages.	C3 (Apply)	2,4
	CLO3: Demonstrate the equivalence of automata, regular expressions, and grammars.	C4 (Analyze)	2,3
	CLO4: Analyze the limitations of finite automata and context-free grammars.	C4 (Analyze)	2,3
	CLO5: Apply automata theory concepts in basic compiler components such as lexical analysis and pattern matching.	C3 (Apply)	2,4,5
Text Book(s)	Peter Linz, "An Introduction to Formal Languages and Automata", 6th Ed., Jones & Bartlett Publishers, 2016. ISBN: 1284077241, 978-1284077247		
Reference Material	Daniel I. A. Cohen, "Introduction to computer theory", 2nd Ed., Wiley India Pvt. Limited, 2007. ISBN 0471137723, 9780471137726		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Course introduction and background. Review of sets, formal languages, grammars, and automata. Introduction to the Chomsky hierarchy. Basics of Kleene closure and its properties.	Linz: Chapter 1
Week 2	Introduction to finite automata. Types of finite automata: acceptors and transducers. Deterministic finite automata (DFA) and their representation using transition graphs, tables, and functions. Introduction to regular languages.	Linz: Chapter 2
Week 3	Construction and analysis of deterministic finite automata (DFA). DFA constructions for union, concatenation, complement, and reversal of regular languages. Introduction to cross product construction for DFA.	Linz: Chapter 2 Quiz1
Week 4	Introduction and motivation for nondeterministic finite automata (NFA). Representation of NFA using transition graphs, tables, and transition functions. λ-NFA (epsilon-NFA) and its conversion to NFA.	Linz: Chapter 2 Quiz2
Week 5	Conversion of NFA to DFA. Equivalence of DFA and NFA. Minimization of deterministic finite automata (DFA).	Linz: Chapter 2
Week 6	Introduction to regular expressions and their formal definition. Operations on regular expressions: union, concatenation, and Kleene star.	Linz: Chapter 3 Quiz3
Week 7	Conversion of regular expressions to finite automata. Introduction to	Linz: Chapter 3

	conversion of finite automata to regular expressions (state elimination method). Equivalence of regular expressions and finite automata. Applications of regular expressions.	Quiz4
Week 8	Introduction to grammars. Regular grammars: production rules and their classification into right-linear and left-linear grammars. Equivalence of regular grammars with finite automata and regular expressions.	Linz: Chapter 3
Week 9	Finite automata with output (transducers): Moore and Mealy machines and their representation using transition graphs, tables, and transition functions. Equivalence of Moore and Mealy machines.	Linz: Chapter 2
Week 10	Pumping lemma for regular languages and its applications.	Linz: Chapter 4 Quiz5
Week 11	Introduction to pushdown automata (PDA). Motivation for PDA based on limitations of finite automata. Types of pushdown automata (PDA). Formal definition of PDA. Representation of PDA using transition diagrams and transition functions.	Linz: Chapter 7
Week 12	Introduction to context-free grammars (CFG). Components of CFG. Derivations and parse trees. Ambiguity in CFG.	Linz: Chapter 5 Quiz6
Week 13	Simplification of CFG (removal of useless symbols, null productions, and unit productions). Chomsky Normal Form (CNF) and Backus–Naur Form (BNF).	Linz: Chapter 6
Week 14	Equivalence of pushdown automata and context-free grammars. Construction of PDA from CFG and CFG from PDA. Pumping Lemma for context-free languages	Linz: Chapters 7–8 Quiz7
Week 15	Introduction to Turing machines (TM). Motivation and limitations of previous computational models. Formal definition of TM. Representation of TM using transition diagrams and transition functions. Design of Turing machines for simple problems. Turing machines with output.	Linz: Chapter 9 Quiz8
Week 16	Variants of Turing machines. Universal Turing machine. Introduction to decidability and undecidability. Applications of Turing machines in computational theory.	Linz: Chapters 10–12

Course Title	**Information Security**		
Course Code	**CSC-313**		
Credit Hours	3 (3,0)		
Category	Computer Science Core		
Prerequisite	None		
Co-Requisite	None		
Follow-up	None		
Course Introduction	This course provides a broad overview of the threats to the security of information systems, the responsibilities and basic tools for information security, and the levels of training and expertise needed in organizations to reach and maintain a state of acceptable security. It covers concepts and applications of system and data security. Areas of particular focus include secure network design, implementation, and transition issues, and techniques for responding to security breaches.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Explain fundamental concepts of information security (confidentiality, integrity, availability).	C2 (Understand)	1,2
	CLO2: Identify common security threats and vulnerabilities.	C3 (Apply)	3
	CLO3: Apply security measures such as encryption, authentication, and access control.	C3 (Apply)	4,5
	CLO4: Conduct basic security audits and risk assessments	C4 (Analyze)	3,4
	CLO5: Understand legal and ethical issues related to information security.	C2 (Understand)	8,9
Course Description	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.		
Text Book(s)	1. M. Whitman and H. Mattord, Principles of Information Security, 6th edition.		
Reference Material	1. William Stallings, Computer Security: Principles and Practice, 3rd edition. 2. Dieter Gollmann, Computer Security, 3rd edition. 3. William Easttom, Computer Security Fundamentals, 3rd edition.		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to information Security. Confidentiality, Integrity, and Availability (CIA). Malware: Viruses, worms, trojan horses, rootkit, spyware, trap door, logic bomb.	M. Whitman: Chapter 1
Week 2	Attacks: Malware, Security breach, DoS, Web Attack, Cross- site scripting, DNS poisoning, identity theft. DoS Attacks: ICMP/Syn/UDP flood. Security Mechanisms: Encryption, Signing, Authentication, honeypots.	M. Whitman: Chapter 1,2
Week 3	Legal, ethical and professional issues in information security. The functions of and relationships among laws, regulations, and professional organizations in information security. Difference between laws and ethics.	M. Whitman: Chapter 3
Week 4	Identification of major national laws that affect the practice of information security. The role of culture as it applies to ethics in information security.	M. Whitman: Chapter 3

Week 5	Planning for security. Management's role in development, maintenance, and enforcement of information security policy, standards, practices and procedures, and guidelines. Information security blueprints, its major components, and its support for the information security program.	M. Whitman: Chapter 4
Week 6	Institutionalization of organization's policies, standards and practices, using education, training and awareness programs. Contingency planning, and its relation to incidence response planning, disaster recovery planning and business continuity plans.	M. Whitman: Chapter 4
Week 7	Cryptography. Symmetric Cipher Model. Substitution and Transposition techniques. Block, Stream and Product Cipher principles. Shanon's S-P Networks.	M. Whitman: Chapter 8
Week 8	Fiestal Cipher, DES Encryption, DES Decryption, Avalanche Effect, Differential and Linear Cryptanalysis.	M. Whitman: Chapter 8
Week 9	Risk Management and its role in organization. Risk management techniques to identify and prioritize risk factors for information assets.	M. Whitman: Chapter 5
Week 10	Assessment of risk based on the likelihood of adverse events and the effects on information assets when events occur. Documentation of the results of risk identification.	M. Whitman: Chapter 5
Week 11	Security Technology. Firewalls and its processing modes. Packet filtering, Application gateways, Circuit gateways, MAC layer firewalls, hybrids. Firewall architectures. VPNs.	M. Whitman: Chapter 6
Week 12	Security Technology: Intrusion Detection and Prevention Systems. IDPS Detection Methods: Signature based, statistical anomaly based, Stateful protocol analysis based, Log file monitors.	M. Whitman: Chapter 7
Week 13	IDPS response behavior. Selecting IDPS approaches and products. Strengths and limitations of IDPS. Deployment and implementation of IDPS. NIDPS, HIDPS.	M. Whitman: Chapter 7
Week 14	Implementing Security. Converting blueprint to a project plan. Organizational considerations to be addressed by project plan. Significant role and importance of the project manager in the success of information security project. The need for professional project management for complex projects. Technical strategies and models for implementing the project plan. Non-technical problems faced by the organization in times of rapid change.	M. Whitman: Chapter 10
Week 15	Security and Personnel: Where and how the information security function is positioned within organizations. Understanding the issues and concerns about staffing the information security function. Knowing about the credentials that professionals in the information security field can acquire.	M. Whitman: Chapter 11
Week 16	Security and Personnel: Recognizing how an organization's employment policies and practices can support the information security effort. Special security precautions necessary for nonemployees. The need for the separation of duties. The special requirements needed for the privacy of personnel data.	M. Whitman: Chapter 11

Course Title	**Cloud Computing**		
Course Code	**CSC-314**		
Credit Hours	3 (3,0)		
Category	Computer Science Core		
Prerequisite	Operating Systems, Computer Networks		
Co-Requisite	None		
Follow Up	Distributed Systems, Big Data Analytics		
Course Introduction	This course provides a comprehensive understanding of cloud computing principles, architectures, and services. It covers cloud service models (IaaS, PaaS, SaaS), deployment strategies, virtualization, cloud security, and emerging trends. The course emphasizes analytical thinking, design of cloud-based solutions, and exposure to real-world applications through case studies and demonstrations.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1:Explain core concepts, characteristics, and service models of cloud computing	C2 (Understand)	1,2
	CLO2: Analyze cloud architectures, virtualization, and deployment models	C4 (Analyze)	2,3
	CLO3:pply cloud technologies and platforms to solve computing problems	C3 (Apply)	3,5
	CLO4: Evaluate cloud security, privacy, and performance issues	C5 (Evaluate)	4
	CLO5: Design cloud-based solutions for real-world applications	C6 (Create)	3,5
Text Books(s)	1. Thomas Erl, Ricardo Puttini, Zaigham Mahmood, Cloud Computing: Concepts, Technology & Architecture, Pearson		
Reference Book(s)	1. Kris Jamsa, Cloud Computing, Jones & Bartlett 2. Rajkumar Buyya, Mastering Cloud Computing, McGraw Hill 3. Official AWS Documentation		

Weekly Lesson Plan		
Week No.	**Course Content (CLO)**	**Assessment**
Week 1	Introduction to Cloud Computing, Evolution, and Characteristics	Erl et al.: Chapter 1-2 Jamsa: Chapter 1
Week 2	Cloud Service Models (SaaS, PaaS, IaaS)	Erl et al.: Chapter 3 Buyya: Chapter 1 Quiz 1
Week 3	Cloud Deployment Models (Public, Private, Hybrid)	Erl et al.: Chapter 4 Jamsa: Chapter 2 Assignment 1
Week 4	Virtualization Concepts and Types	Buyya: Chapter 2 Erl et al.: Chapter 5 Quiz 2
Week 5	Virtual Machines and Infrastructure (IaaS)	Erl et al.: Chapter 6 AWS Documentation (EC2 Overview)
Week 6	Platform as a Service (PaaS) and Development Tools	Erl et al.: Chapter 7 AWS Documentation Quiz 3

Week 7	Cloud Architecture (Basic)	Erl et al.: Chapter 8-9 Assignment 2
Week 8	Cloud Architecture (Advanced)	Review Weeks 1-7
Week 9	Data Management, Load Balancing in Cloud	Buyya: Chapter 5 Jamsa: Chapter 6 Quiz 4
Week 10	Containerization (Docker, Kubernetes Basics – Conceptual)	Docker Documentation Kubernetes Documentation (Introductory Sections) Assignment 3
Week 11	Cloud Pricing Models and Service Metrics	Erl et al.: Chapter 10 AWS Pricing Documentation Quiz 5
Week 12	Cloud Security Fundamentals (CIA Triad)	Jamsa: Chapter 9 Buyya: Chapter 10
Week 13	Encryption, Threats, and Security Mechanism	Erl et al.: Chapter 11 AWS Security Best Practices
Week 14	Designing Cloud-Based Solutions (Case Studies)	Erl et al.: Chapter 12 Selected Cloud Case Studies Quiz 6
Week 15	Emerging Trends (Mobile Cloud, Big Data, SDN)	Recent Research Articles and Industry Reports Presentations
Week 16	Course review and future trends	Comprehensive Course Review

COMPUTER SCIENCE ELECTIVE: 24 (24, 0)

Sr.	Code	Course Title	Prerequisite	Cr. Hrs.
1.	CSE-320	Parallel & Distributed Computing	CSC-213	3 (3,0)
2.	CSE-321	Software Quality Engineering	CSC-214	3 (3,0)
3.	CSE-322	Machine Learning	CSI-230, GE-164	3 (3,0)
4.	CSE-323	Web Technologies	CSC-210	3 (3,0)
5.	CSE-324	Internet of Things and It's Applications		3 (3,0)
6.	CSE-325	Cryptography		3 (3,0)
7.	CSE-326	Compiler Construction	CSC-312	3 (3,0)
8.	CSE-421	Computer Vision	CSI-230, GE-164	3 (3,0)
9.	CSE-422	Mobile Application Development	CSC-210	3 (3,0)
10.	CSE-423	Human Computer Interaction		3 (3,0)
11.	CSE-424	Software Construction & Development	CSC-210	3 (3,0)
12.	CSE-425	Blockchain and Web 3 Security		3 (3,0)
13.	CSE-426	Big Data Analytics		3 (3,0)
14.	CSE-427	Game Design and Development	CSC-210	3 (3,0)
15.	CSE-428	DevOps Principles and Practices	CSC-210	3 (3,0)
16.	CSE-429	Low-code/No-code Application Development		2 (2,0)
17.	CSE-429-L	Low-code/No-code Application Development Lab		1 (0,1)

Course Title	**Parallel and Distributed Computing**		
Course Code	**CSE-320**		
Credit Hours	3 (3,0)		
Category	Computer Science Elective		
Prerequisites	CSC-213 Operating Systems		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course introduces the fundamentals of parallel and distributed computing, focusing on the theory, techniques, and tools to develop high-performance and scalable applications. Topics include Introduction to Parallel & Distributed Computing; Parallel Computing Fundamentals; Distributed Computing Fundamentals; GPU Architectures & Programming Model; Cloud and Edge Computing; and Distributed Data Processing.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: To describe the principles of parallelism, distributed systems, and scalability.	C2 (Understand)	1, 2
	CLO2: To implement algorithms for parallel and distributed environments.	C3 (Apply)	3, 4
	CLO3: To explore modern tools and frameworks used in distributed computing systems.	C4 (Explore)	3, 4
	CLO4: To analyze performance bottlenecks and optimize parallel and distributed systems.	C2 (Analyze)	8, 9
	CLO5: Acquire the knowledge and skills needed to create parallel and distributed environments	C6 (Acquire)	5, 6, 7
Text Book(s)	1. Parallel and Distributed Computing, Ajit Singh, independently published,2022 2. GPU Architecture: The Ultimate Guide to Building High-Performance Computing Systems, Cobbs Walker, independently published,2024 3. Parallel and High-Performance Computing, Robey, R., Zamora, Y., Manning, 2021. 4. Distributed and Cloud Computing: From Parallel Processing to the Internet of Things, Hwang, K., Fox, C. G., Dongarra, J. J., Morgan Kaufmann, 2011.		
Reference Material	1. Distributed Systems: Concepts and Design, Coulouris, G., Dollimore, J. & Kindberg, Addison Wesley, Pearson, 2011.		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to Parallel & Distributed Computing: Concepts of Parallelism & Distribution	Singh: Chapter 1
Week 2	Shared Memory vs. Distributed Memory Systems; Applications & Use Cases of Parallel & Distributed Computing	Singh: Chapter 2
Week 3	Parallel Computing Fundamentals: Types of Parallelism	Assignment 01
Week 4	Data, Task, & Pipeline Parallelism; Multithreading & Multiprocessing	Walker: Chapter 1
Week 5	Parallel Programming Models (OpenMP, MPI, MapReduce).	Singh: Chapter 4
Week 6	Distributed Computing Fundamentals: Principles of Distributed Systems: Consistency	Walker: Chapter 3 Quiz1
Week 7	Availability, and Partition (CAP) Tolerance Theorem	Walker: Chapter 4

Week 8	Communication Protocols: Message Passing & RPCs	
Week 9	Distributed Algorithms: Leader Election, Consensus, Fault Tolerance & Security in Distributed Systems.	Walker: Chapter 3 Assignment 02
Week 10	GPU Architectures & Programming Model: GPU Concepts & Thread Engine, Characteristics, Multi-GPU Platforms & MPI, and GPU Accelerated Platforms	Robey et al.: Chapter 4 Singh: Chapter 3
Week 11	GPU Programming Model: Programming Framework, Code Structure, Optimizing, Synchronization, Asynchronous, and Computing Application Parallelization.	Robey et al.: Chapter 5 Quiz2
Week 12	Cloud & Edge Computing: Cloud computing paradigms (IaaS, PaaS, SaaS); Distributed computing on cloud platforms (AWS, Azure, GCP)	Robey et al.: Chapter 6
Week 13	Distributed computing using local computing devices; Introduction to Fog & Edge Computing & their applications.	Assignment 03
Week 14	Distributed Data Processing: Hadoop Distributed File Systems (HDFS)	Quiz3
Week 15	Big Data Frameworks: Apache Hadoop, Spark & Strom	Walker: Chapter 6,7
Week 16	MapReduce Programming Model & YARN.	Walker: Chapter 6,7

Course Title	**Software Quality and Testing**		
Course Code	**CSE-321**		
Credit Hours	3 (3,0)		
Category	Computer Science Elective		
Prerequisite	CSC-214 Software Engineering		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course is designed to introduce students to the core principles, methods, and practices of Software Quality Assurance. Through a blend of theory and hands-on exercises, learners will gain a deeper understanding of what it takes to ensure software products meet the required standards of quality before they reach the end-user.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Explain software testing and quality assurance principles.	C2 (Understand)	1
	CLO2: Apply the concepts of software verification and validation.	C3 (Apply)	4
	CLO3: Apply various software testing techniques in development project.	C3 (Apply)	6
	CLO4: Design automated test cases.	C5(Design)	4,5
Text Book(s)	1. Daniel Galin, Software Quality Assurance, From theory to implementation, 2014, Pearson, ISBN: 978-81-317-2395-1 2. Paul Jorgensen, Software Testing, A Craftsman's Approach, 4th Ed. CRC Press, Taylor and Francis Group, 2015		
Reference Material	1. Bernard Homes, Fundamentals of Software Testing, ISTE, Wiley, 2012 2. Software Engineering, "Ian Sommerville, 9th Edition, Addison-Wesley, 2011		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to Software Quality Assurance, The Meaning of Quality: Defining Quality, The Quality Challenge, Why is quality important. Context and Motivation, What is software failure? Why do software fail? Quality hazards in Industry	Galin: Chapter 1
Week 2	What is Software Quality, The components of Software quality assurance system	Galin: Chapter 2 Quiz 1
Week 3	Quality of software Requirements	Futrell: Chapter 16 Assignment 1
Week 4	Software quality factors	Galin: Chapter 4
Week 5	Software quality metrics	Galin: Chapter 22
Week 6	Software Reviews: Buddy Checking, Desk Checking, Walkthroughs and Inspections: Overview and Introduction	Galin: Chapter 8 Quiz 2
Week 7	Software Inspections	Galin: Chapter 8
Week 8	Software Configuration Management, Software quality assurance plan	Futrell: Chapter 30 Futrell: Chapter 31
Week 9	Software Testing: Introduction and its basics, Who should do the testing? Test case designing	Desikan: Chapter 1

	Creating Software Testing Plan, Test reporting, When to stop testing?	
Week 10	Software Testing Techniques vs. Software Testing Strategies, Software Testing Techniques: Introduction to White Box Testing and its techniques: Basis Path Testing, Condition Testing, Data Flow Testing, Loop Testing Basis Path Testing: Flow Graph Notation, Cyclomatic Complexity, Basis Paths identification, designing the test cases against each path	Desikan: Chapter 3
Week 11	Adequacy of White-box and Black-box Testing, Introduction to Black-box Testing, introduction to different techniques: Graph, Based Testing, Equivalence, Partitioning, Boundary Value Analysis, Comparison Testing, Orthogonal Array Testing etc., some examples and exercises	Desikan: Chapter 4 Quiz 3
Week 12	Software Testing Strategies: Introduction, V-Model and W-Model, Introduction to different strategies like Unit testing, Integration Testing, System Testing, Acceptance Testing, Verification and Validation and System Testing, Acceptance Testing vs. Verification and Validation Automated Testing: introduction, pluses and minuses of Manual Testing, Why Automated Testing	Desikan: Chapter 5 Assignment 3
Week 13	Introduction of the Testing Tools	Quiz 4
Week 14	Organization Standardization, why and how., Capability Maturity Model: Definition, History, Structure, CMM levels, Key Process Areas.	Godbole: Chapter 8 Assignment 4
Week 15	Transiting to CMM-I Models, Process Maturity levels, Choosing a CMM-I Models Representation, Comparison of SW CMM and CMM-	Godbole: Chapter 9
Week 16	Case Studies and Research Topics	Recent papers

Course Title	**Machine Learning**		
Course Code	**CSE-322**		
Credit Hours	3 (3,0)		
Category	Computer Science Elective		
Prerequisite	GE-164 Probability & Statistics, CSI-230 Linear Algebra		
Co-Requisite	None		
Follow Up	None		
Course Introduction	A mathematically involved introduction to Machine Learning. Present basic machine learning concepts. Present a range of machine learning algorithms along with their strengths and weaknesses. Apply machine learning algorithms to solve problems of moderate complexity.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Explain core concepts of Machine Learning.	C2 (Understand)	1,2
	CLO2: Apply machine learning algorithms to real-world datasets.	C3 (Apply)	3,4,5
	CLO3: Analyze the performance of ML models using evaluation metrics.	C4 (Analyze)	2,3
	CLO4: Implement ML solutions using modern toolkits.	C3 (Apply)	3,4,5
	CLO5: Discuss ethical considerations while applying ML to societal problems.	C2 (Discuss)	8,9
Text Book(s)	1. Deep Learning: Foundations and Concepts, 1st Edition, Christopher Bishop and Hugh Bishop, Springer, 2024, ISBN: 978-3-031-45467-7		
Reference Material	1. Machine Learning: A Probabilistic Perspective, 1st Edition, Kevin P. Murphy, MIT Press, 2022, ISBN: 978-0-262-36930-5. 2. Reinforcement Learning: An Introduction, 2nd Edition, Richard S. Sutton and Andrew G. Barto, MIT Press, 2018, ISBN: 978-0-262-03924-6. 3. Data Mining and Machine Learning: Fundamental Concepts and Algorithms, 2nd Edition, Mohammed J. Zaki, Wagner Meira, Jr., Cambridge University Press, 2020, ISBN: 978-1108473989.		

Weekly Lesson Plan		
Week No.	**Course Content**	**Assignments/ Readings**
Week 1	Introduction to Machine Learning	Bishop: 1.1, 1.3 Python Tutorial
Week 2	Curve Fitting and Regularization	Bishop: 1.2 Quiz 1, Assignment 1
Week 3	Probability and Statistics	Bishop: 2.1, 2.2
Week 4	Gaussian Distribution and Maximum-Likelihood Estimation	Bishop: 2.3, 3.2, 4.1 Quiz 2
Week 5	Matrix Calculus and Universal Approximation Theorem	Bishop: Appendix A, C Assignment 2
Week 6	Loss and Activation Functions for ML	Bishop: 6.4 Quiz 3
Week 7	Neural Networks	Bishop: 6.3
Week 8	Gradient Descent	Bishop: 7 Quiz 4
Week 9	Automatic Differentiation	Bishop: 8.2 Assignment 3
Week 10	Regularization Methods in ML	Bishop: 9 Quiz 5
Week 11	Convolutional Neural Networks	Bishop: 10
Week 12	Transformers	Bishop: 12.1 – 12.3 Quiz 6
Week 13	Clustering	Bishop: 15.1 Zaki 15.1 Assignment 4
Week 14	Principal Component Analysis	Bishop: 16.1 Quiz 7
Week 15	Generative Adversarial Networks	Bishop: 17
Week 16	Reinforcement Learning	Sutton: 1, 5, 6, 13 Quiz 8

Course Title	**Web Technologies**		
Course Code	**CSE-323**		
Credit Hours	3 (3,0)		
Category	Computer Science Elective		
Prerequisite	CSC-210 Data Structures		
Co-Requisite	None		
Follow Up	None		
Course Introduction	Detailed study of the fundamental concepts and technologies used in the development of modern web applications. The course covers the structure and functioning of the World Wide Web, client-server architecture, and both front-end and server-side programming. Emphasis is placed on designing and implementing interactive web interfaces using HTML, CSS, and JavaScript, as well as developing dynamic web applications using Java-based technologies such as Servlets and JSP. The course also introduces key programming paradigms, data handling techniques, and web application architecture, focusing on the design, functionality, and efficiency of web-based systems.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Demonstrate understanding of web architecture and protocols	C2 (Understand)	2,3
	CLO2: Design and implement responsive web pages using HTML, CSS, and JavaScript.	C6 (Create)	4,5
	CLO3: Use modern web development tools and frameworks effectively.	C3 (Apply)	5,10
	CLO4: Work collaboratively to develop and test web applications.	C3 (Apply)	6,4
	CLO5: Communicate design decisions through documentation and presentations.	C4 (Analyze)	7,3
Text Book(s)	1. Schildt, Herbert. Java: the complete reference. McGraw-Hill Education Group, 2014. 2. Team, Wrox Author, et al. Professional Java Server Programming J2EE, (2001).		
Reference Material	1. Hall, Marty, and Brown, Larry. Core Servlets and JavaServer Pages, Prentice Hall Professional, 2004 2. Bergsten, Hans, and Inotai László. JavaServer pages. Sebastopol, CA: O'reilly, 2001.		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to the course, significance of web technologies, assessment and marking criteria, overview of the Internet and World Wide Web, static vs dynamic web applications	Handouts (www.w3schools.com)
Week 2	Introduction to HTML (basic structure, elements, attributes, headings, paragraphs, links, images), HTML tables and forms (form controls, input types), Introduction to CSS (inline, internal, external CSS, selectors, colors, fonts, box model)	Handouts (www.w3schools.com) Assignment 1
Week 3	Introduction to JavaScript (role in web applications, variables, data types, operators), event handling, DOM manipulation basics, form validation techniques	Handouts (www.w3schools.com) Quiz 1
Week 4	Fundamentals of Object-Oriented Programming (classes, objects, methods, constructors, encapsulation)	Schildt: Chapter 6-7
Week 5	Inheritance and polymorphism concepts, method overloading and overriding	Schildt: Chapter 8-9

Week 6	Exception handling mechanisms, introduction to multithreading and thread lifecycle	Schildt: Chapter 10 Quiz 2
Week 7	Introduction to streams, byte and character streams, file handling (reading and writing data to files)	Schildt: Chapter 13 Assignment 2
Week 8	Object streams, introduction to Java Database Connectivity (JDBC), database interaction basics	Schildt: Chapter 13 Wrox et al.: Chapter 4 Quiz 3
Week 9	Fundamentals of client-server architecture, request-response mode	Schildt: Chapter 23
Week 10	Development of a basic server, development of a client application	Schildt: Chapter 23
Week 11	Handling multiple client requests concurrently, client-server communication using specific protocols	Schildt: Chapter 11 Quiz 4
Week 12	Introduction to web applications, layered web application architecture	Wrox et al.: Chapter 5
Week 13	Overview of server-side programming, introduction to Java Servlets, servlet lifecycle	Schildt: Chapter 35 Wrox et al.: Chapter 6 Quiz 5
Week 14	Server-side administration using log files, error handling and custom error pages, use of initialization parameters	Schildt: Chapter 35 Wrox et al.: Chapter 6 Semester Project
Week 15	Stateless vs stateful protocols, techniques for maintaining user state (session management, cookies), servlet chaining, applet-servlet communication	Wrox et al.: Chapter 7 Quiz 6
Week 16	Introduction to JavaServer Pages (JSP), JSP lifecycle, directives, expressions, scriptlets, implicit objects, actions, JavaBeans, introduction to Model-View-Controller (MVC) architecture	Wrox et al.: Chapter 10-11 Case-study

Course Title	**Internet of Things and It's Applications**		
Course Code	**CSE-324**		
Credit Hours	3 (3,0)		
Category	Computer Science Elective		
Prerequisite	None		
Co-Requisite	None		
Follow-up	None		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Differentiate between the layers of the IoT stack based on the TCP/IP model, demonstrating familiarity with the key technologies and protocols employed at each layer.	C4 (Analyze)	2,3
	CLO2: Implement IoT applications by integrating sensors and communication modules.	C3 (Apply)	3,4,5
	CLO3: Understand performance, scalability, security, and privacy issues in IoT solutions across different application domains.	C2 (Explain)	1
	CLO4: Build a functional IoT system by prototyping, programming, and analyzing data, ensuring the system meets specified project requirements.	C5 (Design)	4,5
	CLO5: Evaluate the role of big data, machine learning, and cloud/fog/edge computing in an IoT solution, assessing how each component contributes to the system's functionality and effectiveness.	C6 (Evaluate)	3
Course Description	This course introduces students to the fundamental concepts and applications of the Internet of Things (IoT). It covers the architecture, protocols, and standards that form the backbone of IoT systems. Students will explore IoT stack layers, ranging from physical connectivity to application layers, and understand how these layers interact in a networked environment. Key topics include IoT applications in smart cities, healthcare, agriculture, and industrial settings, as well as the performance, security, scalability, and privacy challenges in IoT solutions. The course will also delve into advanced topics such as cloud, fog, and edge computing, big data processing, and the role of machine learning in IoT. Through a combination of theoretical lessons and practical exercises, students will gain the skills necessary to design and implement functional IoT systems, evaluate the effectiveness of IoT solutions, and understand how to integrate sensors, communication modules, and data analytics for real-world applications. By the end of the course, students will be capable of building and analyzing IoT systems while addressing critical issues such as security, scalability, and interoperability.		
Text Book(s)	1. Internet of Things: Architectures, Protocols and Standards, by Simone Cirani, Gianluigi Ferrari, Marco Picone, and Luca Veltri, 1st edition, Wiley,2019 2. Internet of Things (IoT): Principles, Paradigms and Applications of IoT by Kamlesh Lakhwani, Hemant Kumar Gianey, Joseph Kofi Wireko, Kamal Kant Hiran, 2020		
Reference Material	1. The Internet of Things" by Samuel Greengard, MIT press, 2015 2. A Reference Guide to the Internet of Things, Bridgera LLC, RIoT, 2017		

Weekly Lesson Plan		
Week No.	**Course Content**	**Assignments/Readings**
Week 1	Introduction to IoT. Definition of IoT, Evolution from WSNs to IoT, IoT ecosystem overview.	Cirani: Chapter 1 (1.1–1.2)

Week 2	IoT Applications and Motivation. Smart homes, cities, grids, IIoT, farming. IoT value across domains.	Cirani: Chapter 1 (1.3)
Week 3	TCP/IP Model Review. OSI vs TCP/IP stack. Layered architecture fundamentals.	Cirani: Chapter 2 (2.1)
Week 4	Physical & Link Layer Technologies. Ethernet, Wi-Fi, IEEE 802.15.4, BLE, ZigBee	Cirani: Chapter 2 (2.1.1, 2.2.2)
Week 5	Network Layer in IoT. IPv4 vs IPv6. 6LoWPAN adaptation.	Cirani: Chapter 2 (2.1.2, 2.2.3)
Week 6	Transport Layer Protocols. TCP vs UDP. Lightweight transport considerations.	Cirani: Chapter 2 (2.1.3, 2.2.4)
Week 7	Application Layer Protocols. HTTP, CoAP, AMQP, SIP. IoT-specific protocol trade-offs.	Cirani: Chapter 2 (2.1.4, 2.2.5)
Week 8	IoT Architecture Design. IP-based IoT architecture. Industrial IoT requirements.	Cirani: Chapter 2 (2.2.1 + Industrial IoT 2.3)
Week 9	Interoperability in IoT. REST architecture. Web of Things (WoT). Resource-oriented design.	Cirani: Chapter 3 (3.1 – 3.4)
Week 10	Messaging & Communication Models. MQTT, Pub/Sub. Session initiation.	Cirani: Chapter 3 (3.5 – 3.6)
Week 11	Performance & Data Formats. Performance evaluation. JSON, SensorML, data modeling.	Cirani: Chapter 3 (3.7 – 3.10)
Week 12	Discoverability in IoT. Service discovery (ZeroConf, UPnP). Scalable discovery architectures.	Cirani: Chapter 4
Week 13	Security in IoT. IoT threats. Lightweight cryptography. Key management.	Cirani: Chapter 5 (5.1 – 5.2)
Week 14	Privacy & Authorization. Privacy challenges. IoT authorization frameworks.	Cirani: Chapter 5 (5.3)
Week 15	Cloud, Fog, and Big Data in IoT. Cloud computing. Big data streams. Fog/edge computing.	Cirani: Chapter 6
Week 16	IoT Systems & Practical Implementation. Hardware platforms (Arduino, Raspberry Pi). OS (Contiki, RIOT). IoT testbeds and real-world systems.	Cirani: Chapter 7

Course Title	**Cryptography**		
Course Code	**CSE-325**		
Credit Hours	3 (3,0)		
Category	Computer Science Elective		
Prerequisite	GE-161 Discrete Structures		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course introduces the fundamental principles and techniques of cryptography. It covers classical and modern cryptosystems, cryptanalysis, perfect security, public-key cryptography, and elliptic curve cryptography. The course also explores real-world applications of cryptography in computer security and data protection.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Explain fundamental concepts, terminology, and historical foundations of cryptography	C2 (Understand)	1
	CLO2: Analyze the security and cryptanalysis of classical and modern cryptosystems	C4 (Analyze)	2,3
	CLO3: Apply cryptographic techniques to solve practical security problems	C3 (Apply)	3
	CLO4: Evaluate strengths and weaknesses of cryptographic algorithms and attacks	C5 (Evaluate)	4
	CL05: Design secure cryptographic solutions for real-world applications	C6 (Create)	3,5
Text Books(s)	4. Johannes Buchmann, Introduction to Cryptography 5. Jonathan Katz, Yehuda Lindell, Introduction to Modern Cryptography		
Reference Book(s)	1. William Stallings, Cryptography and Network Security 2. Douglas Stinson, Cryptography: Theory and Practice		

Weekly Lesson Plan		
Week No.	**Course Content (CLO)**	**Assessment**
Week 1	Introduction, Basic Terminology, Cryptosystem Concepts	Buchmann: Chapter 1 Stallings: Chapter 1
Week 2	Historical Cryptosystems (Caesar, Substitution, Vigenère)	Buchmann: Chapter 2 Stinson: Chapter 1 Quiz 1
Week 3	Advanced Classical Ciphers (Hill, Four-square)	Stallings: Chapter 2 Instructor Notes
Week 4	Cryptanalysis of Classical Cryptosystems	Buchmann: Chapter 2 Stinson: Chapter 2 Assignment 1
Week 5	Perfect Security, One-Time Pad	Katz & Lindell: Chapter 2 Buchmann: Chapter 3 Quiz 2
Week 6	Shannon's Theorem, Discrete Logarithm Problem	Katz & Lindell: Chapter 3 Stallings: Chapter 9

		Quiz 3
Week 7	Diffie–Hellman Key Exchange	Buchmann: Chapter 6 Stallings: Chapter 10 Assignment 2
Week 8	Algorithms (Shank's, Pohlig–Hellman)	Stinson: Chapter 7 Research Articles Quiz 4
Week 9	Midterm Exam	Review Previous Chapters
Week 10	Public-Key Cryptography, ElGamal	Buchmann: Chapter 6 Katz & Lindell: Chapter 11
Week 11	Primality Testing, RSA Cryptosystem	Buchmann: Chapter 5 Stallings: Chapter 9
Week 12	Cryptographic Attacks (Broadcast, Wiener, etc.)	Katz & Lindell Chapter 12 Research Papers Assignment 3
Week 13	Knapsack Cryptosystem, Elliptic Curves	Buchmann: Chapter 7 Stinson: Chapter 10 Quiz 5
Week 14	Elliptic Curve Cryptography (ECC)	Buchmann: Chapter 7 Stallings: Chapter 10
Week 15	Applications (Image Encryption, Security Systems)	Recent Journal Articles and Case Studies
Week 16	Final Exam	Comprehensive Course Review

Course Title	**Compiler Construction**		
Course Code	**CSE-326**		
Credit Hours	3 (3,0)		
Category	Computer Science Elective		
Prerequisite	CSC-312 Theory of Automata		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course introduces the fundamental principles and techniques of compiler construction. It covers the complete compilation process from lexical analysis to code generation, including formal language foundations, parsing techniques, syntax-directed translation, and intermediate code generation. Students learn how programming languages are translated into executable form through structured phases of a compiler. The course emphasizes both theoretical foundations and practical implementation aspects, enabling students to design and analyze compiler components such as lexical analyzers, parsers, and semantic analyzers using formal methods and modern tools.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Explain the fundamental concepts of compiler design, including phases of a compiler and language translation process.	C2 (Understand)	1,3
	CLO2: Apply formal techniques to design lexical analyzers, syntax analyzers, and implement basic components of a compiler.	C3 (Apply)	2,3,4,5
	CLO3: Analyze parsing techniques and identify issues involved in compiler implementation and syntax processing.	C4 (Analyze)	3,4,5
Text Book(s)	1. Compilers: Principles, Techniques, and Tools 2nd Edition by Alfred Aho, Jeffrey Ullman, Ravi Sethi, Monica Lam, Addison Wesley, 2006. ISBN: 0321486811, 978-0321486813		
Reference Material	1. Principles of Compiler Design (Express Learning) 1st Edition, Kindle Edition by ITL Education Solutions Limited, Pearson, 2012. ISBN: 978-9332509313 2. Compiler Construction: Principles and Practice 1st Edition by Kenneth C. Louden, Cengage Learning, 1997. ISBN: 0534939724, 978-0534939724		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to Compiler Construction. Importance of compiler design in computing systems. Overview of compilation process. Introduction to machine languages. Expected toolkit and skills after completing the course.	Aho: Chapter 1
Week 2	Phases of a compiler. Detailed architecture of compiler with examples. Relationship between lexical, syntax, semantic analysis, and code generation.	Aho: Chapter 1
Week 3	Introduction to formal languages. Regular languages and expressions. Lexical specifications and role in compiler design.	Aho: Chapter 3 Quiz1
Week 4	Regular expressions (RE). Conversion of RE to NFA. Introduction to NFA and DFA.	Aho: Chapter 3
Week 5	NFA to DFA conversion. DFA optimization. Lookahead and maximum munch principle in lexical analysis.	Aho: Chapter 3 Quiz2
Week 6	Lexical analyzer design. Tokenization process. Specification-dependent lexical analyzer implementation.	Aho: Chapter 3 Quiz3
Week 7	Specification-independent lexical analyzers. Memory optimization techniques in lexical analysis. Efficient scanning strategies.	Aho: Chapter 3 Quiz4

Week 8	Symbol table construction and management. Error handling in lexical analysis. Error detection and recovery. System-level file handling in compiler components.	Aho: Chapters 2–3
Week 9	Syntax analysis (parsing). Context-Free Grammars (CFG). Derivations and parse trees. Ambiguity in grammar.	Aho: Chapter 4
Week 10	Ambiguous vs unambiguous grammars. Removal of ambiguity. Recursive descent parsing and algorithm.	Aho: Chapter 4
Week 11	Limitations of recursive descent parsing. Left recursion and right recursion. Deterministic vs non-deterministic grammars.	Aho: Chapter 4 Quiz5
Week 12	Removal of left recursion (direct and indirect). Left factoring techniques. Grammar transformation for predictive parsing.	Aho: Chapter 4
Week 13	LL(1) parsing. FIRST and FOLLOW sets. Construction of LL(1) parsing tables. Predictive parsing.	Aho: Chapter 4 Quiz6
Week 14	Bottom-up parsing. Shift-reduce parsing concept. LR(0) parsing and canonical collection of LR(0) items.	Aho: Chapter 4 Quiz7
Week 15	SLR(1) parsing and parsing tables. Limitations of LR(0) and SLR(1). CLR(1) parser and canonical LR(1) items.	Aho: Chapter 4 Quiz8
Week 16	LALR(1) parsing and table construction. Introduction to semantic analysis. Attributes and semantic rules. Syntax-directed translation schemes. Intermediate code generation (syntax trees, three-address code).	Aho: Chapters 5–6

Course Title	**Computer Vision**		
Course Code	**CSE-421**		
Credit Hours	3 (3,0)		
Category	Computer Science Elective		
Prerequisite	GE-164 Probability & Statistics, CSI-230 Linear Algebra		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course equips learners with essential image processing concepts, deep learning techniques, and practical skills to solve real-world vision problems.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO 1: Understand and explain the fundamental concepts, techniques, and algorithms used in Computer Vision, including image processing, feature extraction, and object recognition.	C2 (Understand)	1,2
	CLO 2: Implement computer vision algorithms and techniques using modern libraries and frameworks, such as OpenCV, TensorFlow, or PyTorch, to solve real-world image processing problems.	C3 (Apply)	3,4,5
	CLO 3: Evaluate and assess the performance of computer vision models using metrics like accuracy, precision, recall, Intersection over Union (IoU), and mean average precision (mAP).	C4 (Analyze)	2,3
Text Book(s)	1. Foundations of Computer Vision, 1st Edition, Antonio Torralba, Philip Isola, William T. Freeman, MIT Press, 2024, ISBN: 978-0-262-04897-2		
Reference Material	1. Computer Vision: Algorithms and Applications, 2nd Edition, Richard Szeliski, The 2. University of Washington, 2022, ISBN: 978-3-030-34371-2. 3. Multiple View Geometry in Computer Vision, 2nd Edition, Richard Hartley and Andrew 4. Zisserman, ISBN: 978-0-511-81168-5. 5. Digital Image Processing, 4th Edition, Rafael Gonzalez and Richard Woods, 6. Pearson, 2018, ISBN: 978-0-133-35672-4.		

Weekly Lesson Plan		
Week No.	**Course Content**	**Assignments/ Readings**
Week 1	Introduction to Computer Vision	Torralba: Chapter 1, 3, 4 Python Tutorial
Week 2	Image Filtering and Derivatives	Torralba: Chapter 15, 17, 18 Quiz 1, Assignment 1
Week 3	Image Sampling and Multi-scale Representations	Torralba: Chapter 20, 21, 23
Week 4	Machine Learning Basics	Torralba: Chapter 9-11 Quiz 2
Week 5	Neural Networks	Torralba: Chapter 12-13 Assignment 2
Week 6	Deep Learning for Vision - I	Torralba: Chapter 24 Quiz 3
Week 7	Deep Learning for Vision - II	Torralba: Chapter 26
Week 8	Transfer Learning	Torralba: Chapter 37 Quiz 4
Week 9	Representing Images and Geometry	Torralba: Chapter 38 Assignment 3
Week 10	Camera Modelling	Torralba: Chapter 39 Quiz 5
Week 11	Camera Calibration	Torralba: Chapter 39
Week 12	Stereo Vision	Torralba: Chapter 40 Quiz 6
Week 13	Homographies	Torralba: Chapter 41 Assignment 4
Week 14	Single View Metrology and Depth	Torralba: Chapter 42-43 Quiz 7
Week 15	Structure from Motion	Torralba: Chapter 44
Week 16	Optic Flow	Torralba: Chapter 48-49 Quiz 8

Course Title	**Mobile Application Development**		
Course Code	CSE-422		
Credit Hours	3 (3,0)		
Category	Computer Science Elective		
Prerequisite	CSC-210 Data Structures		
Co-Requisite	None		
Follow Up	None		
Course Introduction	A hands-on introduction to mobile application development for the Android platform using Kotlin and Android Studio. Topics include UI design with Jetpack components, activity and fragment lifecycle, data persistence, networking, and publishing apps to the Google Play Store. Emphasis on modern Android development practices and building functional, user-friendly applications.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Recall and describe the Android platform architecture, the role of Android Studio, and the structure of a Kotlin-based Android project.	C1 (Remember)	1,2
	CLO2: Explain the Android Activity and Fragment lifecycle, and describe how UI components interact using layouts, ViewBinding, and event listeners.	C2 (Understand)	1,2
	CLO3: Apply Kotlin language features and Android SDK APIs to build functional multi-screen Android applications.	C3 (Apply)	3,4
	CLO4: Analyze application requirements to select appropriate architecture patterns (e.g., MVVM) and data management strategies including Room and SharedPreferences.	C3 (Apply)	3,4
	CLO5: Evaluate the usability, performance, and code quality of Android applications through testing, debugging, and peer review.	C5 (Evaluate)	4,5
	CLO6: Design and develop a complete Android application that integrates networking, local storage, and a polished UI to address a real-world problem.	C6 (Create)	5,6
Text Book(s)	Dawn Griffiths & David Griffiths, Head First Android Development, 3rd Edition, O'Reilly Media, 2021, ISBN: 978-1492076520		
Reference Material	1. Android Developers Documentation — developer.android.com 2. Kotlin Documentation — kotlinlang.org 3. Big Nerd Ranch Guide: Android Programming, 5th Edition		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to Android & Kotlin: Course overview, Android ecosystem, installing Android Studio, creating first project. Kotlin basics: variables, data types, functions, null safety.	Griffiths: Chapter 1-2
Week 2	Android Project Structure & UI Basics: Project anatomy, AndroidManifest.xml, layouts (LinearLayout, ConstraintLayout), XML UI design, running on emulator and device.	Griffiths: Chapter 3-4 Quiz 1
Week 3	Activities and Intents: Activity lifecycle (onCreate, onStart, onResume, onPause, onStop, onDestroy), explicit and implicit	Griffiths: Chapter 5-6 Assignment 1

	intents, passing data between activities.	
Week 4	ViewBinding and Event Handling: Enabling ViewBinding, replacing findViewById, click listeners, handling user input with EditText, Button, CheckBox, RadioButton, and Toast.	Griffiths: Chapter 7-8 Quiz 2
Week 5	Fragments and Navigation: Fragment lifecycle, adding fragments dynamically, Fragment Manager, Jetpack Navigation Component, NavGraph, and the NavController.	Griffiths: Chapter 9-10
Week 6	RecyclerView and Adapters: Displaying lists with RecyclerView, creating custom Adapters and ViewHolders, handling item click events, DiffUtil for efficient updates.	Griffiths: Chapter 11-12
Week 7	Kotlin Advanced Features: Lambda expressions, higher-order functions, extension functions, data classes, object keyword, companion objects, coroutines introduction.	Griffiths: Chapter 13-14 Assignment 2
Week 8	App Architecture & ViewModel: MVVM pattern, ViewModel and LiveData, separating UI logic from business logic, observing data changes in the UI layer.	Griffiths: Chapter 15-16
Week 9	Data Persistence — SharedPreferences & Room: Storing key-value pairs with SharedPreferences, introduction to Room Database, defining Entities, DAOs, and the Database class.	Griffiths: Chapter 17-18
Week 10	Networking with Retrofit & Coroutines: Making HTTP requests using Retrofit, parsing JSON with Gson/Moshi, Kotlin coroutines with viewModelScope, handling loading and error states.	Griffiths: Chapter 19-20 Assignment 3
Week 11	Images, Permissions, and Media: Loading images with Coil/Glide, requesting runtime permissions, accessing the camera and gallery, handling the permission result callback.	Griffiths: Chapter 21-22 Quiz 4
Week 12	Notifications and Background Work: Creating and displaying notifications, notification channels, introduction to WorkManager for background tasks, scheduling deferred work.	Griffiths: Chapter 23-24
Week 13	Material Design & UI Polish: Material Design components (BottomNavigationView, FloatingActionButton, Snackbar, Card), theming, dark mode support, responsive layouts.	Griffiths: Chapter 25-26 Assignment 4
Week 14	Testing and Debugging: Unit testing with JUnit, UI testing with Espresso, using Logcat and the Android debugger, profiling with Android Profiler, best practices.	Griffiths: Chapter 27-28
Week 15	Publishing & App Review: Preparing app for release (signing APK/AAB), Google Play Store submission process, app review, project presentations.	Griffiths: Chapter 29-30
Week 16	Revision and Project Presentations	Review all chapters; focus on Griffiths: Chapter 15–30 for final exam

Course Title	**Human Computer Interaction**		
Course Code	**CSE-423**		
Credit Hours	3 (3,0)		
Category	Computer Science		
Prerequisites	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	Human-Computer Interaction (HCI) is an interdisciplinary field that focuses on the design, evaluation, and implementation of interactive computing systems for human use. An important goal of HCI is ensuring user satisfaction while interacting with a computing system. Because HCI studies a human and a computer in communication, the multi-disciplinary nature of HCI draws from a number of supporting fields, such as computer graphics, information visualization, programming, psychology, human factors, etc. This course will teach how a human centered approach could lead to effective human-computer interaction.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Understand the fundamental principles and concepts of Human-Computer Interactions	C2 (Understand)	1,2
	CLO2: Apply the principles of good design for people from the perspective of age and disabilities.	C3 (Apply)	3,4
	CLO3: Analyze user needs and behaviors to design effective interactive systems.	C4 (Analyze)	3,4
	CLO4: Evaluate the usability of a medium size software user interface.	C2 (Evaluate)	8,9
	CLO5: Acquire the knowledge and skills needed to create highly usable software systems.	C6 (Acquire)	5,6,7
Text Book(s)	1. *Interaction Design: Beyond Human-Computer Interaction.* By Preece, Jenny, Helen Sharp, and Yvonne Rogers2002 EditionJohn Wiley & Sons 2. *The Design of Everyday Things,* Norman, D. (2013) 3. Human-Computer Interaction. By Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale		
Reference Material	2. Nielsen Norman Group (NNG) – Usability principles, research, and heuristics 3. Interaction Design Foundation – Online HCI/UX courses 4. W3C Web Accessibility Initiative (WCAG) – Accessibility guidelines and standards 5. ACM SIGCHI – Research community for Human-Computer Interaction		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to HCI: Overview of Human-Computer Interaction, its goals, importance, scope, and applications.	Sharp: Chapter 1
Week 2	Human Factors in HCI: Basics of cognitive psychology: human memory, attention, perception, limitations, and ergonomics.	Sharp: Chapter 3
Week 3	Design Principles: Key principles like visibility, feedback, affordances, constraints, mapping, and consistency in UI design.	Sharp: Chapter 1, Norman D: Chapter 3 Assignment 01
Week 4	User-Centered: UCD process, lifecycle, creating personas, task analysis, user goals, and prototyping.	Mastering UI/UX: Chapter 1
Week 5	Interaction Styles: Types of user interfaces: CLI, GUI, menu-based, direct manipulation, and natural interfaces (voice/gesture).	Mastering UI/UX: Chapter 2-3
Week 6	Usability Heuristics: Heuristic principles: Nielsen's 10 rules,	Mastering UI/UX : Chapter

		4
Week 7	Heuristic Evaluations: Shneiderman's rules, and usability goals like effectiveness and satisfaction. Usability testing methods: heuristic evaluation, A/B testing, cognitive walkthroughs.	Mastering UI/UX: Chapter 5
Week 8	Mid Term Exam	
Week 9	Persuasive Technologies, Behavioral Changes, and Anthropomorphism.	Sharp: Chapter 6 Assignment 02
Week 10	Accessibility and Universal Design: Designing for all users, including those with disabilities. Focus on WCAG standards and	Norman D: Chapter 7
Week 11	Face-to-Face Vs Remote Conversations, Co-presence, and Social Engagement.	Sharp: Chapter 5
Week 12	Emerging Trends in HCI: AI in interfaces, AR/VR, gesture-based systems, and brain-computer interfaces.	Norman D: Chapter
Week 13	Ethics in HCI: Privacy, dark patterns, persuasive design, and ethical concerns in AI-based interfaces.	Sharp: Chapter 6
Week 14	Case Studies and Real-World Examples UX analysis of websites/apps	Alan Dix: Chapter 9-10
Week 15	Case Studies and Real-World Examples UX analysis of websites/apps, identifying good/bad UX, and redesign projects	Alan Dix: Chapter 20-21
Week 16	Final Term Exam	

Course Title	**Software Construction and Development**		
Course Code	**CSE-424**		
Credit Hours	3 (3,0)		
Category	Computer Science Elective		
Prerequisite	CSC-210 Data Structures		
Co-Requisite	None		
Follow Up	None		
Course Introduction	Detailed study of the principles and practices involved in the construction and development of modern software systems with a focus on enterprise applications. The course covers object-oriented design, layered software architectures, and the development of web-based systems using Java enterprise technologies. Emphasis is placed on the design and implementation of scalable, secure, and maintainable applications using frameworks, design patterns, and modern development tools. The course also introduces key concepts such as MVC architecture, server-side programming, database integration, web services, transaction management, and deployment, focusing on the structure, quality, and efficiency of software systems.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Understand abstraction, modularity, concurrency, and software design patterns	C2 (Understand)	2,3
	CLO2: Develop optimized, error-free, and reusable code using OOP constructs.	C5 (Design)	4,5
	CLO3: Design and develop a small-scale software system collaboratively.	C5 (Design)	4,6
Text Book(s)	1. Singh, Inderjeet. Designing enterprise applications with the J2EE platform. Addison-Wesley Professional, 2002.		
Reference Material	1. Alur, Deepak, John Crupi, and Dan Malks. Core J2EE patterns: best practices and design strategies. Gulf Professional Publishing, 2003. 2. Sriganesh, Rima Patel, Gerald Brose, and Micah Silverman. Mastering enterprise javabeans 3.0. John Wiley & Sons, 2006.		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Course introduction, overview of software construction and development, revision of Object-Oriented Programming (OOP) concepts	Singh: Chapter 1
Week 2	Software architecture overview (layered, client-server, micro-services), challenges and platforms for enterprise application development	Singh: Chapter 1 Quiz 1
Week 3	Introduction to J2EE, J2EE scenarios, J2EE platform technologies	Singh: Chapter 2
Week 4	J2EE architecture: Client tier and Web tier (Servlets, request-response model)	Singh: Chapter 3
Week 5	Java Servlets in detail, servlet lifecycle, session management, handling error pages	Singh: Chapter 4 Quiz 2
Week 6	Introduction to JavaServer Pages (JSP), JSP lifecycle, directives, expressions, implicit objects, JSP custom tag libraries, defining and using custom tags	Singh: Chapter 4
Week 7	Enterprise JavaBeans (EJB) tier, introduction to business logic layer, introduction to layered architecture	Singh: Chapter 4-5 Quiz 3

Week 8	Model-View-Controller (MVC) architecture and its implementation in web applications	Singh: Chapter 4-5 Assignment 1
Week 9	Introduction to Struts Framework, Struts architecture, controller flow	Handouts
Week 10	Struts components: ActionServlet, Action, ActionForm, practical demonstration	Handouts Assignment 2
Week 11	Database integration: JDBC review, Pre-compiled queries and stored procedures execution	Singh: Chapter 6 Quiz 4
Week 12	Introduction to ORM (e.g., Hibernate), integrating with Enterprise Information Systems	Singh: Chapter 6-7
Week 13	Building RESTful web services, API design principles, JSON/XML data exchange	Handouts Assignment 3
Week 14	Transaction management, security concepts	Singh: Chapter 7 Quiz 5
Week 15	Authentication and authorization (including JWT token-based security)	Singh: Chapter 8 Handouts Semester Project
Week 16	Filters and interceptors, internationalization and localization, logging and exception handling strategies	Singh: Chapter 8 Handouts

Course Title	**Blockchain and Web3 Security**		
Course Code	**CSE-425**		
Credit Hours	3 (3,0)		
Category	Computer Science Elective		
Prerequisites	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This is a beginner-level course that focuses on the foundational technologies behind blockchain. We will cover the concepts of distributed ledger, consensus mechanisms, authentication techniques, and relevant protocols. The course will provide case studies of blockchain applications such as cryptocurrencies, and B2B/B2C/C2C scenarios. The course will also provide hands-on experience with building and deploying smart contracts. It provides participants with a comprehensive understanding of blockchain technology, including its principles, platforms, and applications, to prepare them for developing decentralized solutions.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Describe various software engineering processes and activities.	C2 (Understand)	1, 2
	CLO2: Apply knowledge of software engineering appropriate to the discipline, particularly in the modeling, design, testing and deployment of software systems.	C3 (Apply)	3, 4
	CLO3: Analyze and solve small scale Software Engineering problems	C4 (Analyze)	3, 4
	CLO4: To understand and analyze the positive and negative implications of using blockchain technology in various industries.	C2 (Understand)	8, 9
	CLO5: To be able to explain the various applications of blockchain.	C6 (Explain)	5, 6, 7
Text Book(s)	Andreas Antonopoulos (2017) Mastering Bitcoin (free), https://unglueitfiles.s3.amazonaws.com/ebf/05db7df4f31840f0a873d6ea14dcc28d.pdf P. Kravchenko and others (2018) Blockchain and Decentralized Systems in 3 volumes Cheng, S. (2024). *Web 3.0: Concept, content and context* (pp. 1-217). Springer. Joseph Bambara and others (2018) Blockchain, A practical Guide to Developing Business, Law and Technology solutions Blockchain Experts: Book added to LMS		
Reference Material	https://www.youtube.com/watch?v=U2P_Epjcx1I&list=PL1Xmyl4aKTRjZTZSA_gVpk2xf6QKr_QyZ The Bitcoin Developer Guide: https://bitcoin.org/en/developer-guide Andreas Antonopoulos and Gavin Wood (2019) Mastering Ethereum. Arvind Narayanan (2016), Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction Kindle Edition.		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to Blockchain	Antonopoulos: Chapter 1
Week 2	Introduction to Ethereum.	Antonopoulos: Chapter 2 Assignment 1
Week 3	Introduction to Blockchain Tokens & Smart Contract.	Andreas Antonopoulos Chapter 3

Week 4	Introduction to Tezos Blockchain & Tokenization.	Bambara: Chapter 3,4 Assignment 2
Week 5	Ethereum Platform.	Antonopoulos: Chapter 4
Week 6	Decentralized Applications (dApps), Decentralized Finance (DeFi)	Antonopoulos: Chapter 6 Quiz 1
Week 7	Regulatory and Legal Considerations.	Antonopoulos: Chapter 7
Week 8	Blockchain Security	Antonopoulos: Chapter 7
Week 9	Privacy and Identity on Blockchain.	Antonopoulos: Chapter 7
Week 10	Scalability and Interoperability, Blockchain Use Cases.	P. Kravchenko: 2,3 Assignment 3
Week 11	Introduction to Web3 Security, Smart Contract Integrity	Cheng et al.: Chapter 1-2
Week 12	Digital Asset Custody, Decentralized Infrastructure Risks	Cheng et al.: Chapter 3
Week 13	DApp Frontend & Middleware Security, Regulatory Compliance & AML	P. Kravchenko: Chapter 4,5 Quiz 2
Week 14	Zero-Knowledge Proofs & Privacy, Incident Response & Recovery	Cheng et al., Chapter 5
Week 15	Blockchain and the Future	Narayanan: Chapter 4
Week 16	Emerging Blockchain Technologies & Job Prospects	Book 2: Chapter 4

Course Title	**Big Data Analytics**		
Course Code	**CSE-426**		
Credit Hours	3 (3,0)		
Category	Computer Science Elective		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course is to help students learn, understand, and practice applications for big data analytics and machine learning approaches. Students will get hands-on experience in applying big data technologies and machine learning techniques to industry applications.		
Course Learning Outcomes (CLO)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1:Understand the fundamental concepts of Big Data and its programming paradigm.	C3 (Understand)	1
	CLO2: Hadoop/MapReduce Programming, Framework, and Ecosystem	C3 (Apply)	3,4
	CLO3: Apache Spark Programming	C3 (Apply)	5
Text Book(s)	1. Paul Zikopoulos and Chris Eaton; Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data; McGraw-Hill. ISBN: 0071790535		
Reference Material	1. Leskovec, J., Rajaraman, A., & Ullman, J. D. (2020). Mining of massive data sets. Cambridge university press. 2. White, T. (2012). Hadoop: The definitive guide. O'Reilly Media, Inc. 3. Lin, J., & Dyer, C. (2010). Data-intensive text processing with MapReduce. Synthesis Lectures on Human Language Technologies, 3(1), 1-177.		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to Big Data Analytics Concept, History, and Trends	Leskovec: Chapter 1 Assignment 1
Week 2	Big Data Platforms Hadoop, Spark, and NoSQL Stores	White: Chapter 1 Zikopoulos: Chapter 4 Quiz 1
Week 3	Data Store & Processing using Hadoop Introduction to MapReduce	White: Chapter 2 Leskovec: Chapter 2
Week 4	Programming Hadoop Big Data Storage and Analytics	White: Chapter 3 Hadoop: Chapter 6 Assignment 2
Week 5	HDFS NoSQL databases	White: Chapter 3 Zikopoulos: Chapter 5 Quiz 2
Week 6	Big Data Analytics using ML Algorithms Introduction to Predictive Analytics	Leskovec: Chapter 12 Zikopoulos: Chapter 2 Assignment 3
Week 7	Machine Learning Algorithm Types Big Data and ML	Leskovec: Chapter 7, Chapter 12

		Assignment 4
Week 8	Introduction to Recommendation Systems Type of Recommendation System	Leskovec: Chapter 9
Week 9	Recommendation Systems using Big Data Importance of Supervised and Unsupervised Learning	Leskovec: Chapter 7, Chapter 9 Quiz 3
Week 10	Introduction to Supervise Learning Supervised Learning Algorithms	Leskovec: Chapter 12
Week 11	Introduction to Unsupervised Learning Unsupervised Learning Algorithms	Leskovec: Chapter 7 Assignment 6
Week 12	Cases Studies for Supervised and Unsupervised Learning Linked Big Data: Graph Computing and Graph Analytics	Leskovec: Chapter 5, Chapter 10 Quiz 4
Week 13	Big Data Visualization Dimensionality Reduction in Big Data	Leskovec: Chapter 11
Week 14	Big Data Applications in Healthcare, IoT, and Smart Cities Big Data Applications in Healthcare, IoT, and Smart Cities (Cont.)	Zikopoulos: Chapter 6 Quiz 5
Week 15	Case Studies for Big Data in different Industries Research topics in Big Data Analytics	Zikopoulos : Chapter 5–6 (pp.97–176)
Week 16	Research topics in Big Data Analytics (Cont.)	Recent survey papers

Course Title	**Game Design and Development**		
Course Code	**CSE-427**		
Credit Hours	3 (3,0)		
Category	Computer Science Elective		
Prerequisite	CSC-210 Data Structures		
Co-Requisite	None		
Follow Up	None		
Course Introduction	In this course the students explore fundamental game design theory, rapid prototyping techniques, and practical implementation skills to build complete playable game prototypes. Topics include core game mechanics, player input, physics, 2D game development, UI design, audio, basic AI, and game polishing. Emphasis is placed on the iterative design process: design, prototype, playtest, and refine. No prior game development experience required.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Recall and describe the fundamental concepts of game design including core mechanics, game loops, player psychology, and the Unity3D engine interface.	C1 (Remember)	1,2
	CLO2: Explain game development workflows, the Unity3D scene hierarchy, GameObject-Component model, and the role of C# scripting in game behavior.	C2 (Understand)	1,2
	CLO3: Apply C# programming and Unity3D APIs to implement game mechanics, player controls, physics interactions, and user interface elements.	C3 (Apply)	3,4
	CLO4: Analyze a game design concept to identify required systems, decompose them into components, and select appropriate Unity3D patterns for implementation.	C4 (Analyze)	3,4
	CLO5: Design and develop a complete playable game prototype in Unity3D that integrates core mechanics, polished assets, and a documented design rationale.	C6 (Create)	5,6
Text Book(s)	1. Jeremy Gibson Bond, Introduction to Game Design, Prototyping & Development, 2nd Edition, Addison-Wesley / Pearson, 2017, ISBN: 978-0134659862		
Reference Material	4. Unity Documentation — docs.unity3d.com 5. Microsoft C# Documentation — learn.microsoft.com/dotnet/csharp 6. Unity Learn — learn.unity.com (free tutorials)		

Weekly Lesson Plan

Week No.	Course Contents	Assignments/Readings
Week 1	Introduction to Game Design & Unity3D: Course overview, what is game design, installing Unity3D and VS Code, creating a first Unity project, understanding the Unity Editor interface, Scenes, GameObjects, and the Inspector.	Chapter 1: Thinking Like a Designer (pp. 3–26) Chapter 2: The Layered Tetrad (pp. 27–54)
Week 2	Core Game Design Concepts: Game loops, mechanics, dynamics, and aesthetics (MDA framework), player motivation and psychology, prototyping on paper. Reading: Gibson Bond Ch. 1–3.	Chapter 1: Thinking Like a Designer (pp. 3–26) Chapter 2: The Layered Tetrad (pp. 27–54) Chapter 3: The Formal

		Elements of Games (pp. 55–90) Quiz 1
Week 3	Introduction to C# for Unity: Variables, data types, conditionals, loops, methods. Writing your first MonoBehaviour script, understanding Start() and Update(), attaching scripts to GameObjects.	Chapter 23: Hello World: Your First Program (pp. 531–566) Chapter 24: Variables and Components (pp. 567–602) Chapter 25: Boolean Operations and Conditionals (pp. 603–636) Assignment 1
Week 4	GameObjects, Components & Transforms: The Component model, Transform manipulation via script, Instantiate and Destroy, Prefabs, working with the Hierarchy and Project panels.	Chapter 26: Loops (pp. 637–666) Chapter 27: Functions and Parameters (pp. 667–700) Chapter 28: Arrays and Lists (pp. 701–732) Quiz 2
Week 5	Physics & Collision: Rigidbody2D and Rigidbody3D, Colliders, triggers vs. collisions, OnCollisionEnter / OnTriggerEnter, applying forces, introduction to Unity Physics simulation.	Chapter 29: Classes (pp. 733–770) Chapter 30: Prototyping in Unity (pp. 771–808)
Week 6	Player Input & Controls: Unity Input System (new), reading keyboard, mouse, and gamepad input, character movement, camera follow scripts, building a controllable player character.	Chapter 30: Prototyping in Unity (pp. 771–808) Chapter 31: Apple Picker Prototype (pp. 809–848)
Week 7	Game Mechanics Prototyping: Rapid prototyping techniques from Gibson Bond, iterating on a mechanic, playtesting basics, collecting and applying feedback to refine gameplay.	Chapter 6: Prototyping (pp. 139–166) Chapter 7: Playtesting (pp. 167–194) Chapter 8: Iteration and Paper Prototyping (pp. 195–222) Assignment 2
Week 8	2D Game Development: Sprites and SpriteRenderer, Tilemaps, Animator and Animation Clips, 2D physics, building a complete 2D platformer level prototype.	Chapter 31: Apple Picker Prototype (pp. 809–848) Chapter 32: Apple Picker Coding (pp. 849–888) Chapter 33: Bartok Card Game Prototype (pp. 889–928)
Week 9	UI & HUD Design: Unity UI Canvas system, Text, Image, Button, Slider, HealthBar, ScoreDisplay, menu screens, scene transitions with SceneManager.LoadScene.	Chapter 34: Bartok Card Game Coding (pp. 929–974) Chapter 35: Mission Demolition Prototype (pp. 975–1012)

Week 10	Audio & Game Feel: AudioSource and AudioClip, background music, sound effects, basic screen shake and particle effects, using audio to reinforce game mechanics.	Chapter 35: Mission Demolition Prototype (pp. 975–1012) Chapter 36: Mission Demolition Coding (pp. 1013–1050) Chapter 10: The Aesthetics of Games (pp. 253–280) Assignment 3
Week 11	Object-Oriented Design in Games: Inheritance vs. composition in game contexts, interfaces, ScriptableObjects for data-driven design, event systems, and decoupling game systems.	Chapter 29: Classes (pp. 733–770) Chapter 37: Space SHMUP Prototype (pp. 1051–1090) Chapter 4: The Designer's Approach (pp. 91–112) Quiz 3
Week 12	Game State & Persistence: Managing game states (Start, Play, Pause, GameOver), PlayerPrefs for saving high scores and settings, simple save/load systems using JSON serialization.	Chapter 37: Space SHMUP Prototype (pp. 1051–1090) Chapter 38: Space SHMUP Coding (pp. 1091–1130) Chapter 5: Meaningful Decisions & Emergence (pp. 113–138)
Week 13	Enemy AI & Basic Pathfinding: State machines for enemy behavior (Patrol, Chase, Attack), NavMesh basics, line-of-sight detection, building a simple enemy AI for a prototype level.	Chapter 39: Dungeon Delver Prototype (pp. 1131–1170) Chapter 40: Dungeon Delver Coding (pp. 1171–1210) Chapter 9: Challenges and Victory Conditions (pp. 225–252) Assignment 4
Week 14	Playtesting, Debugging & Optimization: Structured playtesting sessions, gathering feedback, Unity Profiler, identifying bottlenecks, draw call reduction, object pooling pattern.	Chapter 7: Playtesting (pp. 167–194) Chapter 8: Iteration and Paper Prototyping (pp. 195–222) Quiz 4
Week 15	Game Polishing & Build Export: Adding juice (animations, particles, sound), UI polish, building for PC/WebGL with Unity Build Settings, preparing a playable build for submission.	Chapter 10: The Aesthetics of Games (pp. 253–280) Chapter 40: Dungeon Delver Coding (pp. 1171–1210)
Week 16	Revision and Game Project Presentations: Students present their final game prototypes; peer feedback and course review.	Review Chapter 1–10 (Design Theory) Review Chapter 23–40 (Unity Prototypes)

Course Title	**DevOps Principles and Practices**		
Course Code	**CSE-428**		
Credit Hours	3 (3,0)		
Category	Computer Science Elective		
Prerequisite	CSC-210 Data Structures		
Co-Requisite	None		
Follow Up	None		
Course Introduction	DevOps Principles and Practices bridges the gap between software development and operations by fostering a culture of collaboration, automation, and shared responsibility. Designed around the CAMS framework (Culture, Automation, Measurement, and Sharing), this course moves beyond traditional silos to teach students how to build resilient, scalable systems that can handle rapid, continuous delivery. Through the study of the "Three Ways" and the implementation of automated CI/CD pipelines, Infrastructure as Code, and container orchestration, students will transform from individual coders into engineers capable of managing the entire software lifecycle in modern, cloud-native environments.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Explain the core principles and cultural shifts necessary for successful DevOps adoption.	C2 (Understand)	1,8
	CLO2: Implement automated pipelines for continuous integration and continuous delivery of software.	C3 (Apply)	2,3
	CLO3: Use industry-standard tools and techniques to build and deploy modern, cloud-based applications.	C3 (Apply)	5,6
	CLO4: Use collaborative tools and scripting languages to foster teamwork in DevOps practices	C4 (Characterize)	9,10
Text Book(s)	Kim, G., Humble, J., Debois, P., & Willis, J. (2021). The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations (2nd Ed.). IT Revolution Press.		
Reference Material	Arundel, J., & Domingus, J. (2022). Cloud Native DevOps with Kubernetes: Building, Deploying, and Scaling Modern Applications in the Cloud (2nd Ed.). O'Reilly Media. Site Reliability Engineering (Google SRE Book).		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	DevOps Intro: Agile vs. DevOps, CAMS Model.	Kim et al.: Part I (Agile, History, and Introduction to CAMS) Arundel et al.: Chapter 1 (What is Cloud Native?)
Week 2	The Three Ways: Flow, Feedback, Continuous Learning.	Kim et al.: Part I (The Three Ways: Principles of Flow, Feedback, and Continual Learning)
Week 3	CI/CD Lifecycle: Theoretical Pipeline Stages.	Project Assignment 1 Kim et al.: Part II, Chapter 6(Customer Delivery Value Stream) & Part III, Chapter 9

		(Create Automated Testing Foundations)
Week 4	Deployment: Blue-Green & Canary Deployment Logic.	Kim et al.: Part III, Chapter 12 (Low-Risk Releases: Architecting for Low-Risk Deployments) Quiz 1
Week 5	IaC Principles: Idempotency & Configuration Drift.	Kim et al.: Part III, Chapter 11 (Enable Automated Provisioning and Configuration Management)
Week 6	Infrastructure Provisioning: Declarative Provisioning.	Arundel et al.: Chapter 3 (Tools of the Trade: Infrastructure as Code foundational concepts)
Week 7	Containerization: Theory of Isolation & Layers.	Arundel et al.: Chapter 4 (Containers: Images, Dockerfiles, and Runtime Isolation) Quiz 2
Week 8	Orchestration: K8s Architecture & Desired State.	Arundel et al.: Chapter 5 (Kubernetes Primitives, Control Plane Architecture, Pods, and Services)
Week 9	Microservices: API Gateways & Service Discovery.	Kim et al.: Part III, Chapter 10 (Architect for Low-Risk Releases and Evolutionary Architecture) Assignment 2 Arundel et al.: Chapter 7 (Microservice Routing and Services)
Week 10	DevSecOps: Shifting Security Left; Vulnerability Theory.	Kim et al.: Part V, Chapter 22 (Information Security into the Value Stream: Shifting Left) Quiz 3
Week 11	Cloud Native: 12-Factor App Methodology.	Arundel et al.: Chapter 2 (Cloud Native App Architecture and The Twelve-Factor Philosophy)
Week 12	Observability: Metrics, Logs, and Tracing Theory.	Kim et al.: Part IV, Chapter 14 (Create

		Telemetry to See and Solve Problems) Arundel et al.: Chapter 13 (Observability: Metrics and Telemetry) Assignment 3
Week 13	Reliability: SLIs, SLOs, and Error Budgets.	Google SRE Book: Chapter 1 (Introduction to SRE) & Chapter 4 (Service Level Objectives) Quiz 4
Week 14	Chaos Engineering: Building Anti-fragile Systems.	Kim et al.: Part IV, Chapter 16 (Enable Feedback So Production Issues Can Be Fixed Safely) Google SRE Book: Chapter 23 (Managing Critical State: Distributed Consensus for Reliability)
Week 15	Collaboration: ChatOps & Post-mortem Culture.	Kim et al.: Part IV, Chapter 15 (Analyze Telemetry to Better Achieve Goals) & Part V, Chapter 19 (Blameless Post-Mortems) Quiz 5
Week 16	Final Project Demos	Final Project Viva

Course Title	**No-Code/Low-Code Application Development**		
Course Code	**CSE429**		
Credit Hours	2 (2,0)		
Category	Computing Elective		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course explores the paradigm shift in software engineering known as the "Abstraction Layer," focusing on the synergy between Large Language Models (LLMs) and No-Code/Low-Code (NCLC) frameworks. As the industry moves toward rapid prototyping and agile deployment, the ability to architect scalable systems without traditional syntax is a critical competency for modern developers and product architects. Throughout the course, students will move from conceptual design to professional-grade deployment by architecting **EduFlow** (a comprehensive educational management system) or any other hands-on project. Students will master the principles of **Systems Architecture** by progressing from visual databases to professional-grade backend environments. While the course begins with the visual simplicity of **Airtable** and **Softr**, it bridges the gap to enterprise-level development by introducing **Supabase** for relational PostgreSQL management, **WeWeb** for sophisticated, logic-heavy frontend engineering, and **Make.com** for complex, multi-step workflow automation.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Architect complex application blueprints by leveraging LLMs to generate technical schemas, logic flowcharts, and system requirements.	C6 (Create)	3,4,5
	CLO2: Engineer scalable relational database architectures by transitioning from visual data modeling in Airtable to PostgreSQL backends in Supabase.	C6 (Create)	3,4,5
	CLO3: Develop high-fidelity, data-driven user interfaces by mapping dynamic data to modular components within Softr and WeWeb.	C3 (Apply)	3,4,5
	CLO4: Orchestrate sophisticated middleware logic and automated workflows using Make.com to synchronize the EduFlow ecosystem.	C3 (Apply)	3,4,5
	CLO5: Execute secure Identity and Access Management (IAM) strategies through Row-Level Security (RLS) and conditional visibility	C3 (Apply)	3,5,8,9
	CLO6: Synthesize the trade-offs of the "Abstraction Layer" to evaluate when to deploy no-code versus low-code solutions for professional scalability.	C5 (Evaluate)	3,8,10
Text Book(s)	2. Low-Code/No-Code: Citizen Developers and the Surprising Future of Business Applications, written by Phil Simon, ISBN-13 979-8985814736. 3. Building Production-Grade Web Applications with Supabase by David Lorenz, ISBN-13: 978-1837630684.		
Course Assessment Strategy	This course adopts a project-based assessment approach in place of a traditional written examination, with the primary objective of evaluating students' ability to design, develop, and deploy a functional application using low-code/no-code tools taught to them in this course. The students (individually) are required to propose and finalize a project idea involving the development of a real-world application. They must design and implement this application using the tools and platforms introduced in the course, progressing through guided milestones throughout the semester under their teacher's supervision. The final developed application serves as the primary basis for evaluation. Student performance will be assessed based on this application. The assessment strategies may include: project demonstration, presentation, and verbal question-and-answer (viva voce).		

Weekly Lesson Plan			
Week No.	**Course Contents**	**Assignment(s)**	**Tool(s)**
Phase 1: AI-Assisted Architecture			
Week 1	The Modern Stack: Intro to the "Abstraction Layer." How to use LLM Personas for system design.	Prompt AI to design the EduFlow database schema. Map it to Airtable for a visual start.	Claude & Airtable
Week 2	Cloud-Native Data: Relational design and data integrity. Using AI for "Mock Data."	Generate 100 mock student/course records using ChatGPT; import them into Airtable.	ChatGPT & Airtable
Week 3	UI/UX Foundations: Component-based design. Mapping data to a UI instantly.	Connect Softr to Airtable. Use AI to generate SEO-friendly course descriptions for your public catalog.	Softr & Airtable
Week 4	Identity & Access (IAM): Theoretical authentication and visibility rules.	Prompt AI for a "Logic Flowchart" of who sees what. Implement these visibility rules in Softr.	Softr & Claude
Phase 2: Automation & "Vibe" Workflows			
Week 5	The CRUD Lifecycle: Software patterns for data manipulation (Create, Read, Update, Delete).	Build a "Student Registration" form. Use AI to troubleshoot any connection errors.	Softr & Airtable
Week 6	Serverless Automations: Event-driven architecture. Designing "Trigger-Action" sequences.	Prompt AI for logic: "Email a PDF invoice on registration." Build this workflow in Make.com.	Make.com & Gmail
Week 7	Computational Logic: Understanding aggregation and relational "Roll-ups."	Build a "Professor Portal." Use Airtable roll-ups to calculate total attendance automatically.	Softr & Airtable
Week 8	Mid-term Milestone: Quality assurance and documentation of your AI-assisted logic.	Deliverable: A functional v1 portal. Submit your "Prompt History" logs to show your work.	All Phase 1 & 2 Tools
Phase 3: Advanced Frontend Logic			
Week 9	Professional Layouts: Migrating to a professional-grade frontend. Intro to Flexbox/CSS.	Rebuild the Student Dashboard for a custom, high-fidelity experience in WeWeb.	WeWeb
Week 10	Database Migration: Moving from "Spreadsheets" to "Databases."	Connect WeWeb to Supabase. Map your student data collection variables for real-time updates.	WeWeb & Supabase
Week 11	Visual Formulas: Translating business requirements into visual logic for apps.	Use AI to generate WeWeb formulas for complex logic (e.g., GPA or late fee calculations).	WeWeb & Claude

Week 12	API Ecosystems: RESTful APIs and JSON. Connecting your app to the outside world.	Use AI to interpret API docs and connect your app to a live Campus News or Weather feed.	WeWeb & Groq
Phase 4: Low-Code Bridge & Deployment			
Week 13	Hybrid Development: Injecting JavaScript for custom behaviors when No-Code hits a wall.	Use AI to write a JS snippet for a custom feature (like a custom date-picker or dark mode toggle).	WeWeb & ChatGPT
Week 14	AI Agent Integration: Building an "Agentic" workflow inside your application.	Build a "Course Advisor Chatbot" inside the app using a free AI API (like Groq).	WeWeb & Groq
Week 15	Security & Optimization: Data privacy audit and performance tuning.	Feed your app logic to AI to find security holes or redundant database steps.	Supabase & AI
Week 16	Full-Stack Deployment: Hosting and the launch lifecycle. Final production demo.	Final Result: A live, production-ready EduFlow Portal built via No-Code and AI.	Supabase & WeWeb

Course Title	**No-Code/Low-Code Application Development Lab**		
Course Code	**CSE-429-L**		
Credit Hours	1 (0,1)		
Category	Computing Elective		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	In the lab course of No-Code/Low-Code Application Development, the labs on topics taught in each week must be designed and conducted. This course explores the paradigm shift in software engineering known as the "Abstraction Layer," focusing on the synergy between Large Language Models (LLMs) and No-Code/Low-Code (NCLC) frameworks. As the industry moves toward rapid prototyping and agile deployment, the ability to architect scalable systems without traditional syntax is a critical competency for modern developers and product architects. Throughout the course, students will move from conceptual design to professional-grade deployment by architecting **EduFlow** (a comprehensive educational management system) or any other hands-on project. Students will master the principles of **Systems Architecture** by progressing from visual databases to professional-grade backend environments. While the course begins with the visual		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Architect complex application blueprints by leveraging LLMs to generate technical schemas, logic flowcharts, and system requirements.	C6 (Create)	3,4,5
	CLO2: Engineer scalable relational database architectures by transitioning from visual data modeling in Airtable to PostgreSQL backends in Supabase.	C6 (Create)	3,4,5
	CLO3: Develop high-fidelity, data-driven user interfaces by mapping dynamic data to modular components within Softr and WeWeb.	C3 (Apply)	3,4,5
	CLO4: Orchestrate sophisticated middleware logic and automated workflows using Make.com to synchronize the EduFlow ecosystem.	C3 (Apply)	3,4,5
	CLO5: Execute secure Identity and Access Management (IAM) strategies through Row-Level Security (RLS) and conditional visibility	C3 (Apply)	3,5,8,9
	CLO6: Synthesize the trade-offs of the "Abstraction Layer" to evaluate when to deploy no-code versus low-code solutions for professional scalability.	C5 (Evaluate)	3,8,10
Text Book(s)	1. Low-Code/No-Code: Citizen Developers and the Surprising Future of Business Applications, written by Phil Simon, ISBN-13 979-8985814736. 2. Building Production-Grade Web Applications with Supabase by David Lorenz, ISBN-13: 978-1837630684.		
Course Assessment Strategy	This course adopts a project-based assessment approach in place of a traditional written examination, with the primary objective of evaluating students' ability to design, develop, and deploy a functional application using low-code/no-code tools taught to them in this course. The students (individually) are required to propose and finalize a project idea involving the development of a real-world application. They must design and implement this application using the tools and platforms introduced in the course, progressing through guided milestones throughout the semester under their teacher's supervision. The final developed application serves as the primary basis for evaluation. Student performance will be assessed based on this application. The assessment strategies may include: project demonstration, presentation, and verbal question-and-answer (viva voce).		

Weekly Lesson Plan			
Week No.	**Course Contents**	**Labs Topics**	**Tool(s)**
Phase 1: AI-Assisted Architecture			
Week 1	The Modern Stack: Intro to the "Abstraction Layer." How to use LLM Personas for system design.	Prompt AI to design the EduFlow database schema. Map it to Airtable for a visual start.	Claude & Airtable
Week 2	Cloud-Native Data: Relational design and data integrity. Using AI for "Mock Data."	Generate 100 mock student/course records using ChatGPT; import them into Airtable.	ChatGPT & Airtable
Week 3	UI/UX Foundations: Component-based design. Mapping data to a UI instantly.	Connect Softr to Airtable. Use AI to generate SEO-friendly course descriptions for your public catalog.	Softr & Airtable
Week 4	Identity & Access (IAM): Theoretical authentication and visibility rules.	Prompt AI for a "Logic Flowchart" of who sees what. Implement these visibility rules in Softr.	Softr & Claude
Phase 2: Automation & "Vibe" Workflows			
Week 5	The CRUD Lifecycle: Software patterns for data manipulation (Create, Read, Update, Delete).	Build a "Student Registration" form. Use AI to troubleshoot any connection errors.	Softr & Airtable
Week 6	Serverless Automations: Event-driven architecture. Designing "Trigger-Action" sequences.	Prompt AI for logic: "Email a PDF invoice on registration." Build this workflow in Make.com.	Make.com & Gmail
Week 7	Computational Logic: Understanding aggregation and relational "Roll-ups."	Build a "Professor Portal." Use Airtable roll-ups to calculate total attendance automatically.	Softr & Airtable
Week 8	Mid-term Milestone: Quality assurance and documentation of your AI-assisted logic.	Deliverable: A functional v1 portal. Submit your "Prompt History" logs to show your work.	All Phase 1 & 2 Tools
Phase 3: Advanced Frontend Logic			
Week 9	Professional Layouts: Migrating to a professional-grade frontend. Intro to Flexbox/CSS.	Rebuild the Student Dashboard for a custom, high-fidelity experience in WeWeb.	WeWeb
Week 10	Database Migration: Moving from "Spreadsheets" to "Databases."	Connect WeWeb to Supabase. Map your student data collection variables for real-time updates.	WeWeb & Supabase
Week 11	Visual Formulas: Translating business requirements into visual logic for apps.	Use AI to generate WeWeb formulas for complex logic (e.g., GPA or late fee calculations).	WeWeb & Claude

Week 12	API Ecosystems: RESTful APIs and JSON. Connecting your app to the outside world.	Use AI to interpret API docs and connect your app to a live Campus News or Weather feed.	WeWeb & Groq
Phase 4: Low-Code Bridge & Deployment			
Week 13	Hybrid Development: Injecting JavaScript for custom behaviors when No-Code hits a wall.	Use AI to write a JS snippet for a custom feature (like a custom date-picker or dark mode toggle).	WeWeb & ChatGPT
Week 14	AI Agent Integration: Building an "Agentic" workflow inside your application.	Build a "Course Advisor Chatbot" inside the app using a free AI API (like Groq).	WeWeb & Groq
Week 15	Security & Optimization: Data privacy audit and performance tuning.	Feed your app logic to AI to find security holes or redundant database steps.	Supabase & AI
Week 16	Full-Stack Deployment: Hosting and the launch lifecycle. Final production demo.	Final Result: A live, production-ready EduFlow Portal built via No-Code and AI.	Supabase & WeWeb

Interdisciplinary / Allied Courses: 12 (12, 0)

Sr.	Code	Course Title	Prerequisite	Cr. Hrs.
1.	CSI-130	Calculus and Analytical Geometry (Mandatory)		3 (3,0)
2.	CSI-230	Linear Algebra (Mandatory)		3 (3,0)
3.	CSI-330	Digital Marketing & E-Commerce		3 (3,0)
4.	CSI-331	Project Management		3 (3,0)
5.	CSI-332	Bioinformatics		3 (3,0)
6.	CSI-333	Geographic Information Systems (GIS) & Spatial Data		3 (3,0)
7.	CSI-334	Cyber Law & Digital Policy		3 (3,0)

Course Title	**Calculus and Analytical Geometry**		
Course Code	**CSI-130**		
Credit Hours	3 (3,0)		
Category	Interdisciplinary / Allied Courses		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course provides a foundational understanding of calculus and analytical geometry, focusing on how mathematical concepts describe change, motion, and spatial relationships. It begins with limits and continuity, which form the basis for defining functions and understanding their behavior. The course then introduces differentiation, exploring how to measure rates of change and apply derivatives to real-world problems such as optimization and curve analysis. Integration is developed as a complementary concept to differentiation, emphasizing accumulation, area under curves, and practical applications. Alongside calculus, the course includes key ideas from analytical geometry, such as equations of lines and planes in three-dimensional space. Overall, the course equips students with essential mathematical tools used in science, engineering, economics, and other applied fields.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Know the concepts and applications of calculus and analytical geometry.	C1 (Know)	1
	CLO2: Describe functions, limit, continuity, chain rule, and related techniques.	C2 (Describe)	1
	CLO3: Identify and solve problems related to differentiation and integration.	C3 (Apply)	1,3
Text Book(s)	1. Howard Anton, Irl C. Bivens and Stephen Davis, Calculus, 11th Edition, Wiley, 2016, ISBN-10: 1119228581, ISBN-13: 978-1119228585.		
Reference Material	1. Thomas and Finney, Calculus and Analytic Geometry, 9th Edition, ISBN-13: 978-0201531749, ISBN-10: 0201531747.		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Functions, Definition. Classification of functions. Domain and range, Inverse of a function.	Anton et al.: Chapter 0 Anton et al.: Exercise set 0.1 [Questions 4 & 7 To 10]
Week 2	Limit and Continuity: Composition of two and three functions. The concept of limit. Limit of a function.	Anton et al.: Exercise set 0.2 [Questions 4 & 27 To 36] Anton et al.: Chapter 1 Quiz 1
Week 3	Graphical limit, Limits- Computational techniques.	Anton et al.: Exercise set 1.1 [Questions 1 To 10 & 13 To 16] Assignment 1

		Anton et al.: Exercise set 1.2 [Questions 1 To 40]
Week 4	Limits at infinity, Continuity-Definition, Graphical Continuity.	Anton et al.: Exercise set 1.3 [Questions 1 To 4 & 9 To 40] Anton et al.: Exercise set 1.5 [Questions 11 To 22 & 29 To 32] Exercise set 1.6 [Questions 1 To 40]
Week 5	Differentiation: Average rate of change, Instantaneous rate of change, Derivative of a function-Definition, Procedure to find the derivative of the function $y=f(x)$.	Anton et al.: Exercise set 2.1 [Questions 11 To 18] Anton et al.: Exercise set 2.2 [Questions 7 To 22]
Week 6	Differentiation of algebraic functions, Product & Quotient Rules Differentiation of Trigonometric Functions.	Anton et al.: Exercise set 2.3 [Questions 1 To 24 & 29 To 32 & 41 To 48] Exercise set 2.4 [Questions 1 To 24 & 27 To 36] Anton et al.: Exercise set 2.5 [Questions 1 To 28]
Week 7	Chain Rule, Implicit Differentiation, Differentiation of Logarithmic Functions.	Anton et al.: Exercise set 2.6 [Questions 1 To 4 & 7 To 58] Anton et al.: Exercise set 3.1 [Questions 1 To 20 & 25 To 28]
Week 8	Differentiation of Exponential and Inverse Trigonometric Functions, Applications of Differentiation	Anton et al.: Exercise set 3.2 [Questions 1 To 30 & 35 To 43] Assignment 2 Anton et al.: Exercise set 3.3 [Questions 15 To 58] Quiz 2
Week 9	Optimization: Increasing and decreasing functions., Concavity.	Anton et al.: Exercise set 4.1

		[Questions 15 To 38]
Week 10	Relative Maxima and Minima, Absolute Maxima and Minima	Anton et al.: Exercise set 4.2 [Questions 3 To 14 & 25 To 50] Assignment 3
Week 11	Indefinite Integral: Techniques of integration. [formulas], Power rule of integration. Integrals of the form $\int [f(x)]^n f'(x)dx$	Anton et al.: Exercise set 4.4 [Questions 7 To 16 & 21 To 43] Quiz 3 Anton et al.: Table 5.2.1
Week 12	Integrals of the form $\int \frac{f'(x)}{f(x)} dx$ Integrals of Trigonometric functions.	Anton et al.: Exercise set 5.2 [Questions 1 To 36]
Week 13	Integrals of Inverse Trigonometric functions, Integrals of exponential functions, Term-by-Term integration of composite functions.	Anton et al.: Exercise set 7.3 [Questions 1 To 52]
Week 14	Integration with the use of trigonometric identities, Integration by substitution.	Anton et al.: Exercise set 7.4 [Questions 1 To 26 & 37 To 48]
Week 15	Integration by parts, Integration by partial fractions.	Anton et al.: Exercise set 5.3 [Questions 1 To 65] Exercise set 7.1 [Questions 1 To 30] Anton et al.: Exercise set 7.2 [Questions 1 To 38 & 47 To 54]
Week 16	Applications of Integration, Definite Integral: Properties of Definite integral, Area under the curve. Area enclosed between curves	Anton et al.: Exercise set 7.5 [Questions 1 To 34 & 39 To 42] Quiz 4 Assignment 4

Course Title	**Linear Algebra**		
Course Code	**CSI-230**		
Credit Hours	3 (3,0)		
Category	Interdisciplinary / Allied Courses		
Prerequisite	None		
Co-Requisite	None		
Follow Up	CSE-322 Machine Learning, CSE-421 Computer Vision		
Course Introduction	This course introduces fundamental concepts of linear algebra, including matrices, vectors, systems of equations, and vector spaces. Students develop analytical and computational skills to solve mathematical problems using techniques such as Gaussian elimination and eigenanalysis. The course also emphasizes applications of linear algebra in computer science, including data representation, algorithms, and modeling.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Understand and perform operations on matrices and vectors, including addition, multiplication, and inverse calculations.	C2 (Understand)	1,2,3
	CLO2: Solve systems of linear equations using matrix methods such as Gaussian elimination and matrix inversion.	C3 (Apply)	2,3
	CLO3: Analyze vector spaces, subspaces, basis, and dimension to understand the structure of linear systems.	C4 (Analyze)	2,3
	CLO4: Apply eigenvalues and eigenvectors in solving problems related to transformations and stability analysis.	C3 (Apply)	2,3,4
	CLO5: Utilize linear algebra concepts to model and solve practical problems in computer science, such as algorithms and data representations.	C6 (Create)	2,3,4
Text Book(s)	1. Howard Anton and Chris Rorres, J., Elementary Linear Algebra. Applications Version, 11th Edition, Wiley & Sons, NJ, 2014, ISBN: 978-1-118-43441-3. 2. David C. Lay, Steven R. Lay, Judi J. McDonald, Linear Algebra and Its Applications, 5th Edition, Pearson, 2015, ISBN-13: 978-0321982384, ISBN-10: 032198238X. 3. Gilbert Strang, Introduction to Linear Algebra, 5th Edition, Wellesley-Cambridge Press, 2016, ISBN-13: 978-0980232776, ISBN-10: 0980232775. 4. Howard Anton, Elementary Linear Algebra, 11th Edition, Wiley, 2013, ISBN-13: 978-0470458211, ISBN-10: 0470458216.		
Reference Material	1. Philip N. Klein, Coding the Matrix: Linear Algebra through Applications to Computer Science, 1st Edition, Newtonian Press, 2013, ISBN-13: 978-0615880990, ISBN-10: 0615880991. 2. 2. David Hill, David Zitarelli, Linear Algebra Labs with MATLAB, 3rd Edition, Pearson, 2003, ISBN-13: 978-0131432741, ISBN-10: 0131432745		

Weekly Lesson Plan		
Week No.	**Course Content**	**Assignments/Readings**
Week 1	Systems of linear equations	Anton: 1.1, 1.2
Week 2	Matrix Arithmetic	Anton: 1.2 Quiz 1
Week 3	Matrix Inverse	Anton: 1.4 – 1.6

Week 4	Diagonal and Triangular Matrices	Anton: 1.7 Quiz 2
Week 5	Matrix Transformations	Anton: 1.8
Week 6	Polynomial Interpolation	Anton: 1.9 Quiz 3
Week 7	Determinants - I	Anton: 2.1
Week 8	Determinants - II	Anton: 2.2 – 2.3 Quiz 4
Week 9	Vector Spaces - I	Anton: 3.1 – 3.2
Week 10	Vector Spaces - II	Anton: 3.3 – 3.5 Quiz 5
Week 11	General Vector Spaces - I	Anton: 4.1 – 4.3
Week 12	General Vector Spaces - II	Anton: 4.4 – 4.6 Quiz 6
Week 13	Basis	Anton: 4.7 – 4.10
Week 14	Eigenvalues and Eigenvectors	Anton: 5.1, 5.2 Quiz 7
Week 15	Inner Product Spaces	Anton: 6.1, 6.2
Week 16	Gram-Schmidt Process & Least Squares Fitting	Anton: 6.3, 6.4 Quiz 8

Course Title	**Digital Marketing & E-Commerce**		
Course Code	**CSI-330**		
Credit Hours	3 (3,0)		
Category	Interdisciplinary / Allied Courses		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course introduces the principles, tools, and technologies of digital marketing and e-commerce in the modern digital economy. Key topics include digital marketing channels, SEO, social media, content and email marketing, online advertising, e-commerce models, electronic payment systems, CRM, and digital analytics. Students will gain practical exposure to industry tools to design and evaluate digital campaigns and e-commerce solutions. Ethical, legal, and security aspects of online transactions and data privacy are also emphasized. By the end of the course, students will be able to apply analytical thinking to develop digital strategies and understand e-commerce systems, preparing them for careers in digital business and entrepreneurship.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Explain key concepts of digital marketing, e-commerce, and digital economy.	C2 (Explain)	1,2
	CLO2: Identify appropriate digital marketing channels and strategies for different business scenarios.	C3 (Apply)	1,2
	CLO3: Develop basic e-commerce solutions including online stores and payment integration concepts.	C3 (Apply)	2,3
	CLO4: Analyze digital marketing campaigns using analytics tools and performance metrics.	C4 (Analyze)	2,3
	CLO5: Evaluate ethical, legal, and security issues in digital marketing and e-commerce systems.	C5 (Evaluate)	6
	CLO6: Apply modern tools and platforms for implementing digital marketing campaigns.	C3 (Apply)	3,5
Text Book(s)	4. Dave Chaffey, Fiona Ellis-Chadwick, Digital Marketing: Strategy, Implementation and Practice, 8th/9th Edition, Pearson Education, 5. Kenneth C. Laudon, Carol Guercio Traver, E-Commerce: Business, Technology, Society, 17th/18th Edition, Pearson Education, 2021/2023, ISBN-13: 978-1292403175 / 978-1292444000.		
Reference Material	1. Damian Ryan, Understanding Digital Marketing: Marketing Strategies for Engaging the Digital Generation, 5th Edition, Kogan Page, 2021, ISBN-13: 978-1789666014. 2. P. N. Bhatia, Fundamentals of Digital Marketing, Pearson Education, 2017, ISBN-13: 978-9332585447. 3. Equinet Academy, Fundamentals of Digital Marketing Workbook, Equinet Academy, 2018.		

Weekly Lesson Plan		
Week No.	**Course Content**	**Assignments/Readings**
Week 1	Introduction to Digital Marketing & E-Commerce, Scope and Applications	Ryan: Chapter 1
Week 2	Digital Economy & E-Commerce Business Models	Ryan: Chapter 2 Quiz 1
Week 3	Digital Marketing Channels Overview & Strategy	Ryan: Chapter 3
Week 4	Search Engine Optimization (SEO)	Ryan: Chapter 4 Assignment 1
Week 5	Social Media Marketing Strategies	Ryan: Chapter 5 Quiz 2

Week 6	Content Marketing & Email Marketing	Ryan: Chapter 6 Quiz 3
Week 7	Online Advertising (PPC, Display Ads)	Ryan: Chapter 7 Assignment 2
Week 8	Conversion Rate Optimization (CRO), Landing Page Design, and Customer Journey Mapping	Ryan: Chapter 8 Mid Exam
Week 9	E-Commerce Platforms & Website Design Basics	Ryan: Chapter 9
Week 10	Electronic Payment Systems & Digital Transactions	Ryan: Chapter 10 Assignment 3
Week 11	Customer Relationship Management (CRM)	Ryan: Chapter 11 Quiz 4
Week 12	Digital Analytics & Performance Measurement	Ryan: Chapter 12 Quiz 5
Week 13	Security, Privacy & Legal Issues in E-Commerce	Ryan: Chapter 13
Week 14	Mobile Commerce & Emerging Trends (AI, Automation)	Ryan: Chapter 14 Quiz 6
Week 15	Integration: Designing Digital Marketing Campaigns & E-Commerce Solutions	Ryan: Chapter 15 Project
Week 16	Advanced Topics: Global E-Commerce, Future Trends, Course Review & Project Presentation	Ryan: Chapter 16

Course Title	**Project Management**		
Course Code	**CSI-331**		
Credit Hours	3 (3,0)		
Category	Interdisciplinary / Allied Courses		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course provides a comprehensive introduction to the principles, practices, and tools used in managing software development projects. It covers the full project management lifecycle, including project initiation, planning, execution, monitoring, and closure, with a focus on software-specific challenges. Students will learn key concepts such as project scope definition, scheduling, cost estimation, risk management, quality assurance, and resource allocation. The course also emphasizes the use of modern project management tools and techniques, along with the preparation and use of essential project documentation such as project plans, schedules, and progress reports. Through case studies and practical exercises, students will develop the ability to manage software projects effectively in team-based environments while adhering to professional standards. This course equips students with the managerial and organizational skills required to successfully deliver software projects in real-world settings.		
Course Learning Outcomes	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Describe Project, Project Management	C1 (Describe)	1,2
	CLO2: Apply the project management tools and techniques	C3 (Apply)	4,5
	CLO3: Learn to document project management documents and the usage of these documents in project activities	C3 (Apply)	4,7
	CLO4: Discuss key principles and common methods for software project management such as scheduling, size estimation, cost estimation, and risk analysis	C2 (Discuss)	3,10
Text Book(s)	A. Guide to the Project Management Body of Knowledge 6th Edition		
Reference Material	1. Essentials of Software Project Management by Richard Bechtold 2. Integrated Approach to Software Engineering by Punkaj Jalote		

Weekly Lesson Plan		
Week No.	**Course Content**	**Assignments/Readings**
Week 1	Introduction, what is a Project? What is Project Management? The Project Life Cycle, Project Stakeholders, Organizational Influences, Project management Process Groups, Project management Knowledge Groups (brief description) Portfolio, Program and Project Management. Project Management vs Operations Management, Organizational Influence and Project Life cycle	Book A: 1.2
Week 2	Business Case, Project Selection Methods (Cost Benefit Analysis, Pay Back Period, Return On Investment)	Reading notes Assignment 1, Quiz 1
Week 3	Project Selection Methods continued (Net Present Value, Internal Rate of Return) Develop Project Charter	Reading notes
Week 4	Develop Project Management Plan Direct and Manage Project Work	Book A: 4.2 Assignment 2, Quiz 2

Week 5	Monitor and Control Project Work Integrated Change Control, Close Project	Book A: 4.3
Week 6	Scope Planning, Collect Requirements Define Scope, Create WBS	Book A: 5.1 Assignment 3, Quiz 3
Week 7	Validate Scope , Control Scope	Book A:5.1
Week 8	Plan Schedule management, Activity Definition, Activity Sequencing	Book A: 6.1, 6.2 Quiz 4
Week 9	Activity Resource Estimating, Activity Duration Estimating, schedule development	Book A: 6.3 Assignment 4
Week 10	Schedule Control, Plan Cost Management, Cost Estimating	Book A: 7.1, 7.2
Week 11	Cost Budgeting, Cost Control	Book A:7.3, 7.4 Assignment 5, Quiz 5
Week 12	Quality Planning	Book A: 8.1
Week 13	Perform Quality Assurance, Perform Quality Control, Human Resource Planning, Acquire Project Team	Book A: 8.2, 8.3 Assignment 6, Quiz 6
Week 14	Develop Project Team, Manage Project Team, Communications Planning, Manage Communication	Book A: 9.4, 9.6, 10
Week 15	Risk Management Planning, Risk Identification, Qualitative Risk Analysis Quantitative Risk Analysis, Risk Response Planning, Risk Monitoring and Control	Book A:11 Assignment 7, Quiz7
Week 16	Project Procurement Management	Book A:12

Course Title	**Bioinformatics**		
Course Code	**CSI-332**		
Credit Hours	3 (3,0)		
Category	Interdisciplinary / Allied Courses		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course introduces computer scientists to Bioinformatics, which uses computer databases to store, retrieve and assist in understanding biological information. In this course students will learn the fundamental concepts of existing in-silico methods to access the wealth of data to answer questions relevant to bioinformatics and is highly hands-on. It will provide knowledge related to databases, sequence storage, retrieval and analysis, similarity and homology, creating alignments, local and global alignment, pairwise and multiple sequence alignments, sequence profiling, phylogenetic analysis, dot matrix plots, dynamic programming algorithm, word (k-tuple) methods, substitution matrices PAM and BLOSUM, scoring algorithms, gap penalties, online tools i.e. BLAST and FASTA structures.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Illustrate the basic concepts of bioinformatics and its significance in biological data analysis.	C2 (Understand)	1,2
	CLO2: Implement the computational methodologies and utilize the algorithms employed for biological data interpretation.	C3 (Apply)	3,4
	CLO3: Analyze various sequence-based algorithms to solve problems of proteomics and genomics.	C4 (Analyze)	3,4
Text Book(s)	1. Arthur M. Lesk, Introduction to Bioinformatics, Oxford University Press.		
Reference Material	1. Jin Xiong, Essential Bioinformatics, Cambridge University Press. 2. Yadav Neelam, A Handbook of Bioinformatics, Animal Publications Pvt. Ltd. 3. Krawetz Stephen A., Introduction to Bioinformatics: A Theoretical and Practical Approach, Humana Press.		

Weekly Lesson Plan

Week No.	Course Contents	Assignments/Readings
Week 1	Introduction to Bioinformatics; History of Bioinformatics. Overview of Course Contents – Section I (Genomics); Overview of Course Contents – Section II (Proteomics)	Lesk: Chapter 1-2
Week 2	Need for Bioinformatics – Section I (Genomics); Need for Bioinformatics – Section II (Proteomics). Applications of Bioinformatics – Section I (Genomics); Applications of Bioinformatics – Section II (Proteomics)	Lesk: Chapter 1-2 Assignment 1
Week 3	Frontiers In Bioinformatics – Section I (Genomics); Frontiers In Bioinformatics – Section II (Proteomics). Central Dogma of Molecular Biology; Cell Structure and Localizations	Lesk: Chapter 1 Lesk: Chapter 4 Appendix: Molecular Biology Primer (pp. 467–490) Quiz 1
Week 4	Nucleotides: DNAs & RNAs; Amino Acids: Genes & Proteins Cell Transcription in Eukaryotes; Cell Transcription in Prokaryotes	Appendix: Molecular Biology Primer (pp. 467–

		490) Lesk: Chapter 3
Week 5	Protein Translations in Eukaryotes; Ribosomes, tRNAs & Codon Table. Biological Databases: NCBI & Gene Bank	Appendix: Molecular Biology Primer (pp. 467–490) Lesk: Chapter 3 Lesk: Chapter 5 Assignment 2
Week 6	Biological Sequences Alignments: Pairwise Sequence Alignments – Section I (Genes). Biological Sequences Alignments: Pairwise Sequence Alignments – Section II (Proteins)	Lesk: Chapter 6
Week 7	Pairwise Sequence Alignments – Section III (Peptides); Dot Plots: Gaps, Miss-Matches and Penalties. Sequence Similarity v/s Sequence Identity; Dynamic Programming and Sequence Alignments.	Lesk: Chapter 6 Quiz 2
Week 8	Global Sequence Alignment: Needleman-Wunsch Sequence Alignment Algorithm with Examples. Local Sequence Alignment: Smith-Waterman Sequence Alignment Algorithm with Examples.	Lesk: Chapter 6
Week 9	BLAST Algorithm: Section I (Genes), BLAST Algorithm: Section II (Proteins)	Lesk: Chapter 7
Week 10	FASTA Algorithm: Section I (Genes), FASTA Algorithm: Section II (Proteins)	Lesk: Chapter 7 Assignment 3
Week 11	Genetic Mutations: Types and Examples, Scoring Alignments; Scoring Matrices.	Lesk: Chapter 6 Lesk: Chapter 8
Week 12	PAM Matrices; BLOSUM Matrices, Multiple Sequence Alignments (MSAs); MSAs: Progressive Alignments.	Lesk: Chapter 6 Lesk: Chapter 9 Quiz 3
Week 13	Multiple Sequence Alignments (MSAs); MSAs: Iterative Alignments, Multiple Sequence Alignments (MSAs); MSAs: Center Star Algorithm.	Lesk: Chapter 9 Assignment 4
Week 14	Multiple Sequence Alignments (MSAs); MSAs: Phylogenetic Alignments, Profile Alignments: Position-Specific Based Profiling	Lesk: Chapter 8
Week 15	Position-Probability Based Profiling; Position-Relativity Based Profiling, Protein Structures: Primary, Secondary, Tertiary & Quaternary.	Lesk: Chapter 9 Lesk: Chapter 10 Quiz 4
Week 16	Protein-Protein Interactions; Protein-Peptide Interactions Drug-Targets and Repurposing Tools; Revision.	Lesk: Chapter 4 Lesk: Chapter 10

Course Title	**Geographic Information Systems (GIS) & Spatial Data**		
Course Code	**CSI-333**		
Credit Hours	3 (3,0)		
Category	Interdisciplinary / Allied Courses		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course is designed to familiarize BS Computer Science students with the fundamental concepts of Geographic Information Systems (GIS) and spatial data analysis. It enables students to understand spatial data models, GIS design principles, and the role of data structures in managing geospatial information. The course covers methods of spatial data acquisition, various sources of geospatial data, techniques for querying and analyzing spatial datasets, and effective visualization of geographic information. Students will also develop an understanding of map projections and coordinate systems, and their application in accurate map creation and spatial problem-solving. The course integrates computational thinking with geospatial technologies to prepare students for real-world applications in diverse domains.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Explain fundamental concepts of GIS and spatial data models Identify sources and methods of spatial data acquisition.	C2 (Explain)	1,2
	CLO2: Apply spatial data structures and querying techniques. Analyze spatial data using basic GIS operations (buffer, overlay, proximity).	C3 (Identify)	1,2
	CLO3: Visualize spatial data using appropriate cartographic techniques.	C3 (Apply)	2,3
	CLO4: Demonstrate understanding of map projections and coordinate systems. Use GIS tools/software for spatial data processing and analysis.	C4 (Analyze)	2,3
	CLO5: Explain fundamental concepts of GIS and spatial data models (vector and raster).	C3 (Apply)	3,4
	CLO6: Identify sources and methods of spatial data acquisition. Apply spatial data structures and querying techniques.	C2 (Explain)	1,2
	CLO7: Analyze spatial data using basic GIS operations (buffer, overlay, Visualize spatial data using appropriate cartographic techniques. Demonstrate understanding of map projections and coordinate systems.	C3 (Apply)	3,4
Text Book(s)	1. Geographic Information Science and Systems, (1st edition, 2015), P. A. Longley, Michael Goodchild, David Maguire, David Rhind 2. GIS Fundamentals (latest edition, 2019), Paul Bolstad		
Reference Material	1. Fundamentals of GIS, (latest edition, 2015), Michael Demers, Wiley 2. Essentials of GIS, (latest edition, 2011), Campbell, Shin 3. Geospatial Analysis, (latest edition, 2015), Paul A. de Smith, Michael F. Goodchild, Paul Longley		

Weekly Lesson Plan		
Week No.	**Course Content**	**Assignments/Readings**
Week 1	Introduction to GIS, components, applications, history, and development	Longley et al.: Chapter 1
Week 2	Spatial data types, representation, data models (vector & raster)	Longley et al.: Chapter 3 Quiz 1

Week 3	Vector data model, topology, geodatabases	Longley et al.: Chapter 4 Assignment 1
Week 4	Raster data model, raster analysis, DEMs	Longley et al.: Chapter 5 Quiz 2
Week 5	Spatial data structures (R-tree, Quadtrees), file formats	Longley et al.: Chapter 6
Week 6	Spatial data acquisition: GPS, Remote Sensing	Longley et al.: Chapter 6 Quiz 3
Week 7	Sources of spatial data (OSM, satellite imagery, government datasets)	Longley et al.: Chapter 7 Assignment 2
Week 8	Spatial querying (attribute & spatial queries), selection, SQL	Longley et al.: Chapter 10 Quiz 4
Week 9	Spatial analysis: buffer, overlay, proximity, network analysis	Longley et al.: Chapter 11
Week 10	Data visualization, cartography, map design principles	Longley et al.: Chapter 12 Quiz 5
Week 11	Map projections, distortions, coordinate systems	Longley et al.: Chapter 2 Assignment 3
Week 12	Datum transformations, georeferencing, coordinate conversion	Longley et al.: Chapter 2 Quiz 6
Week 13	GIS software: QGIS/ArcGIS practical exercises	Longley et al.: Chapter 10-12 Assignment 4
Week 14	Advanced analysis and applications (urban planning, disaster management, environmental)	Longley et al.: Chapter 13 Quiz 7
Week 15	Case studies & integrated GIS projects	Longley et al.: Chapter 11-13
Week 16	Advanced issues in GIS (data quality, organizational & human issues, future of GIS, analytical modeling recap), Review of all GIS concepts, project presentations,	Longley et al.: Chapter 14

Course Title	**Cyber Law & Digital Policy**		
Course Code	**CSI-334**		
Credit Hours	3 (3,0)		
Category	Interdisciplinary / Allied Courses		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course provides a comprehensive understanding of the legal, regulatory, and governance frameworks shaping the digital world. It examines the intersection of computing technologies with law and public policy, focusing on issues such as cybercrime, data governance, digital rights, platform regulation, and emerging challenges in areas like artificial intelligence and digital surveillance. The course places particular emphasis on Pakistan's legal and policy landscape, including the Prevention of Electronic Crimes Act, Electronic Transactions Ordinance, and the National Cyber Security Policy, while also drawing comparisons with international frameworks. Through critical analysis of laws, policies, and real-world case studies, students will develop the ability to evaluate regulatory approaches and understand the societal, ethical, and legal implications of digital technologies.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Explain foundational concepts of cyber law, digital policy, and internet governance.	C2 (Understand)	2,8
	CLO2: Analyze cybercrimes, regulatory challenges, and policy implications in digital environments.	C4 (Analyze)	3,8
	CLO3: Apply legal, ethical, and policy frameworks to address digital governance and data-related issues.	C3 (Apply)	4,9
	CLO4: Evaluate national and international digital policies, laws, and regulatory frameworks.	C5 (Evaluate)	3,8
	CLO5: Assess emerging digital policy challenges (AI governance, platform regulation, data sovereignty)	C5 (Evaluate)	8,10
Text Book(s)	1. Cyber Law, Privacy, and Security: Concepts, Methodologies, Tools, and Applications, Information Resources Management Association (USA), 4th Edition, Information Science Reference, IGI Global, 2019, ISBN-10: 1668431149, ISBN-13: 978-1668431146		
Reference Material	1. Tari Schreider, Cybersecurity Law, Standards and Regulations, 2nd Edition, Rothstein Publishing, 2020, ISBN-10: 1944480560, ISBN-13: 978-1944480561 2. Muhammad Azhar Ghani, Cyber Laws in Pakistan, ISBN-13: 979-8632952255 3. Nisha Dhanraj Dewani, Zubair Ahmed Khan, Aarushi Agarwal, Mamta Sharma, Shaharyar Asaf Khan, Handbook of Research on Cyber Law, Data Protection, and Privacy, IGI Global, 2022, ISBN-10: 2021027667, ISBN-13: 1799886433		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to Cyber Law & Digital Policy	Schreider: Chapter 1
Week 2	Introduction to Cyber Law & Digital Policy	Schreider: Chapter 1 Quiz 1
Week 3	Internet Governance & Digital Ecosystems	Assignment-1
Week 4	Legal Frameworks in Pakistan (ETO, PECA, Policy Bodies)	Handouts on ETO, PECA, Policy Bodies Quiz 2

Week 5	Global Digital Policies (GDPR, cross-border data laws)	Schreider: Chapter 3
Week 6	Cybercrime & Regulatory Challenges	Schreider: Chapter 3
Week 7	Policy Responses to Cybercrime (case studies)	Case Study Handouts, Quiz-3
Week 8	Data Governance & Privacy Policies	Schreider: Chapter 3, Assignment-2
Week 9	Platform Regulation (Social Media, Big Tech)	Schreider: Chapter 3
Week 10	Case Study-1	Handouts on case study
Week 11	Digital Forensics & Legal Evidence	Schreider: Chapter 4 Quiz-4
Week 12	Cybersecurity Policy & National Strategies	Handouts Schreider: Chapter 2
Week 13	Intellectual Property & Digital Economy	Handouts Assignment-3
Week 14	Ethics in Digital Policy & Governance	Handouts Quiz-5
Week 15	Emerging Issues: AI Regulation, Data Sovereignty, Digital Surveillance	Schreider: Chapter 7 Project Evaluation
Week 16	Policy Analysis Presentations / Case Study-2/ Projects	Handouts on case study

GENERAL EDUCATION: 34 (30, 4)

Sr.	Code	Course Title	Sub – Category	Cr. Hrs.
1.	GE-160	Applications of Information & Communication Technologies		3 (2,1)
2.	GE-161	Discrete Structures	Quantitative Reasoning – I	3 (3,0)
3.	GE-162	Functional English		3 (3,0)
4.	GE-163	Introduction to Economics	Social Science	2 (2,0)
5.	GE-164	Probability & Statistics	Quantitative Reasoning – II	3 (3,0)
6.	GE-165	Expository Writing		3 (3,0)
7.	GE-166	Pakistan Studies		2 (2,0)
8.	GE-167	Fehm-e-Quran – I		1 (0,1)
9.	GE-168	Fehm-e-Quran – II		1 (0,1)
10.	GE-260	Civics and Community Management		2 (2,0)
11.	GE-261	Ideology and Constitution of Pakistan		2 (2,0)
12.	GE-262	Applied Physics	Natural Science	3 (2,1)
13.	GE-263	Professional Practices	Arts and Humanities	2 (2,0)
14.	GE-264	Islamic Studies		2 (2,0)
15	GE-265	Entrepreneurship		2 (2,0)

Course Title	**Applications of Information & Communication Technologies**		
Course Code	**GE-160**		
Credit Hours	3 (2,1)		
Category	General Education		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	The course is designed to provide students with an exploration of the practical applications of Information and Communication Technologies (ICT) and software tools in various domains. Students will gain hands-on experience with a range of software applications, learning how to leverage ICT to solve daily life problems, enhance productivity and innovate in different fields. Through individual and interactive exercises and discussions, students will develop proficiency in utilizing various software related to ICT.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Explain the fundamental concepts, components, and scope of ICT.	C2 (Explain)	1,2
	CLO2: Identify uses of various ICT platforms and tools for different purposes.	C4 (Identify)	1,2,3
	CLO3: Apply ICT platforms and tools for different purposes to address basic needs in different domains of daily, academic, and professional life.	C3 (Apply)	3,4,5
	CLO4: Understand ethical and legal considerations in use of ICT platforms and tools.	C2 (Understand)	1,2,3
Text Book/ Reference Material	1. "Discovering Computers" by Vermaat, Shaffer, and Freund. 2. Deborah Morley and Charles S. Parker, Understanding Computers: Today and Tomorrow, 16th edition, Cengage Learning, 2016, ISBN-13: 978-1337251853 3. "Computing Essentials" by Morley and Parker. 4. "GO! With Microsoft Office" Series by Gaskin, Vargas, and McLellan. 5. "Exploring Microsoft Office" Series by Grauer and Poatsy. 6. "Technology in Action" by Evans, Martin and Poatsy. 7. Livesley, Robert Kenneth. An introduction to automatic digital computers. Cambridge University Press, 2017. 8. Joan Lambert, Curtis Frye, Microsoft Office 2019 Step by Step, First Edition. ISBN: 978-1-50-930597-1.		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Course Introduction: Motivation, Significance and Future Prospects. Introduction to the World of Computers. Computers in Your Life. What Is a Computer and What Does It Do? Computers to Fit Every Need. Operating System: Explore Task Manager of Microsoft ® Windows. Using Linux Operating System using Virtual Box. Changing Drives, Deleting, Copying, Moving files using command line interface in Linux.	Morley et al. Chapter 1
Week 2	Introduction to the World of Computers. Computer Networks and the Internet. Computers and Society. Basic ICT Productivity Tools: Effective use of popular search engines to explore WWW, Formal communication tools and etiquettes (Gamail, Microsoft Outlook, etc.)	Morely et al.: Chapter 1 Assignment 1

Week 3	The System Unit: Processing and Memory. Data and Program Representation. Inside the System Unit. How the CPU Works. Making Computers Faster and Better Now and in the Future. Microsoft Word: Basic Editing. Character and Paragraph Formatting. Managing Text Flow. Creating Tables. Working with Themes Style Sets, Backgrounds, Quick Parts, and Text Boxes. Using Illustrations and Graphics. Formatting a CV.	Morely et al.: Chapter 2 Lab Manual – V. Quiz 1
Week 4	Microsoft Excel – I: Working with Microsoft excel. Using Office Backstage. Using basic Formulas and Functions. Formatting Cells, Ranges, and Worksheets.	Lab Manual – VI. Assignment 2
Week 5	Storage: Storage Systems Characteristics. Hard Drives. Optical Discs and Drives. Flash Memory Storage Systems. Other Types of Storage Systems. Microsoft Excel – II: Managing Worksheets. Working with Data and using Advanced Formulas. Securing and Sharing Workbooks. Creating Charts adding pictures and shapes to a Worksheet.	Lab Manual – VII. Morely et al.: Chapter 3
Week 6	Microsoft PowerPoint & Primitive Graphics Design Tools: PowerPoint essentials and presentation basics. Working with Text and Designing a Presentation. Adding Tables to Slides. Using Charts in a Presentation. Creating SmartArt Graphics. Creating Graphics and Adding to a Presentation. Using Animation and Multimedia. Securing and Sharing a Presentation. Input and Output: Keyboards. Pointing and Touch Devices. Scanners, Readers, and Digital Cameras. Audio Input. Audio Output. Display Devices. Printers.	Morely et al.: Chapter 4 Lab Manual – VIII. Quiz 2
Week 7	System Software: Operating Systems and Utility Programs: System Software vs. Application Software. The Operating System. Operating Systems for Personal Computers. Operating Systems for Servers. Operating Systems for Mobile Devices. Operating Systems for Larger Computers. Utility Programs. The Future of Operating Systems. Microsoft Access – I: Database Essentials. Creating Database Tables. Working with Tables and Database Records. Modifying Tables and Fields. Creating Forms and Reports.	Lab Manual – IX. Morely et al.: Chapter 5 (page 174 to 207)
Week 8	Computer Networks: What Is a Network? Networking Applications. Network Characteristics. Data Transmission Characteristics. Networking Media. Data Transmission Characteristics. Networking Media. Microsoft Access – II: Creating and Modifying Queries. Using Controls in Reports and Forms.	Morely et al.: Chapter 7 Lab Manual – X Assignment 3
Week 9	The Internet and the World Wide Web: Evolution of the Internet. Getting Set Up to Use the Internet. Searching the Internet. Google Workspace (Google Docs, Sheets, Slides),	Morely et al.: Chapter 8
Week 10	Google Drive, Dropbox (cloud storage and file sharing), Google Drive (Cloud storage with Google Docs integration) and Microsoft OneDrive (Cloud storage with Microsoft integration),	https://drive.google.com/drive/home https://www.dropbox.com/ https://onedrive.live.com/login Quiz 3
Week 11	Evernote (Note-taking and organization applications) and OneNote (Microsoft's digital notebook for capturing and organizing ideas),	https://onenote.cloud.microsoft/
Week 12	Video conferencing (Google Meet, Microsoft Teams, Zoom, etc.), social media applications (LinkedIn, Facebook, Instagram, etc.)	https://meet.google.com/landing Assignment 4
Week 13	ICT in Education: Working with learning management systems (Moodle, Canvas, Google Classrooms, etc.), Sources of online education courses (Coursera, edX, Udemy, Khan Academy, etc.), Interactive multimedia and virtual classrooms	https://moodle.com/ https://www.instructure.com/canvas

		https://www.coursera.org/
Week 14	ICT in Health and Well-being: Health and fitness tracking devices and applications (Google Fit, Samsung Health, Apple Health, Xiaomi Mi Band, Runkeeper, etc.), Telemedicine and online health consultations (OLADOC, Sehat Kahani, Mahram, etc.)	https://oladoc.com/ https://www.marham.pk/ Quiz 4
Week 15	ICT in Personal Finance and Shopping: Online banking and financial management tools (JazzCash, Easypaisa, Zong PayMax, 1Link and MNET, Keenu Wallet, etc.), E-commerce platforms.	https://www.jazzcash.com.pk/ https://1link.net.pk/
Week 16	Digital Citizenship and Online Etiquette: Intellectual property and copyright issues, Ensuring originality in content creation by avoiding plagiarism and unauthorized use of information sources, Content accuracy and integrity (ensuring that the content share through ICT platforms is free from misinformation, fake news, and manipulation).	Review of the course

Course Title	**Discrete Structures**		
Course Code	**GE-161**		
Credit Hours	3 (3,0)		
Category	General Education		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course provides a foundation in discrete mathematical structures essential to computer science, covering formal logic, set theory, functions, relations, proof techniques, counting, induction, graphs, and trees. It emphasizes mathematical reasoning, abstraction, and problem-solving to model and analyze computational systems, while linking theory to applications in algorithms, data structures, databases, artificial intelligence, and software correctness.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Explain fundamental concepts of discrete structures, including logic, sets, functions, relations, graphs, and trees.	C2 (Understand)	1,3
	CLO2: Apply logical reasoning, proof techniques, and discrete structures to model and solve computing problems.	C3 (Apply)	1,2,3,4
	CLO3: Analyze and differentiate discrete structures and evaluate their relevance in computer science applications.	C4 (Analyze)	1,2,3,4
Text Book(s)	Discrete Mathematics and Its Applications 8th Edition by Kenneth Rosen., McGraw Hill Education, 2018. ISBN: 1260091996, 978-1260091991		
Reference Material	Discrete Mathematical Structures 6th Edition by Bernard Kolman, Robert Busby, Sharon C. Ross., Pearson, 2008. ISBN 0132297515, 978-0132297516		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Course introduction and motivation. Importance of Discrete Structures in Computer Science. Applications in algorithms, databases, artificial intelligence, and software engineering. Overview of course contents and learning strategies. Introduction to propositional logic: propositions, truth values, logical connectives, and compound propositions.	Rosen: Chapter 1
Week 2	Construction and evaluation of truth tables. Logical implication, biconditional, and translation of natural language statements into symbolic form. Logical equivalence, tautology, contradiction, and contingency.	Rosen: Chapter 1
Week 3	Logical equivalence using truth tables and standard laws of logic. Simplification of logical expressions. Introduction to predicate logic: predicates, propositional functions, and domain of discourse.	Rosen: Chapter 1–2 Quiz1
Week 4	Universal and existential quantifiers. Negation of quantified statements. Nested quantifiers and interpretation in natural language.	Rosen: Chapter 2
Week 5	Set theory fundamentals: sets, universal set, subsets, cardinality. Power sets and properties. Set operations: union, intersection, difference, and complement. Algebra of sets and identities.	Rosen: Chapter 2 Quiz2
Week 6	Set equality and verification of set identities using Venn diagrams and membership tables. Characteristic vectors and representation of sets as bit-vectors. Applications of sets in computing.	Rosen: Chapter 2 Quiz3
Week 7	Functions: ordered pairs, tuples, and Cartesian products. Definition and representation of functions. Types of functions (injective, surjective, bijective). Inverse and numeric functions with applications.	Rosen: Chapter 2–3 Quiz4

Week 8	Relations: definition and notation. Representation using matrices and directed graphs. Properties of relations: reflexive, symmetric, antisymmetric, and transitive.	Rosen: Chapter 9
Week 9	Composition of relations and closures. Equivalence relations and equivalence classes. Partitioning of sets. Partial order relations and posets.	Rosen: Chapter 9
Week 10	Sequences and summations. Arithmetic and geometric progressions. Recursive definitions. Summation notation, linearity, and evaluation of common and geometric sums.	Rosen: Chapter 3 Quiz5
Week 11	Counting principles: introduction and applications. Sum rule, product rule, complement rule, and inclusion–exclusion principle.	Rosen: Chapter 6 Quiz6
Week 12	Permutations and combinations. Binomial coefficients and applications in counting problems.	Rosen: Chapter 6
Week 13	Mathematical proofs: rules of inference. Direct proof, proof by contrapositive, and proof by contradiction.	Rosen: Chapter 1–3 Quiz7
Week 14	Mathematical induction: principle of induction and applications. Strong induction. Introduction to graph theory: terminology, types, Handshaking Lemma, and graph representations.	Rosen: Chapter 5
Week 15	Walks, paths, and cycles. Connected and strongly connected graphs. k-connected graphs and degree concepts. Havel–Hakimi theorem.	Rosen: Chapter 5 Quiz8
Week 16	Special classes of graphs: complete graphs, paths, cycles, stars, wheels, n-cubes, and bipartite graphs. Trees and their properties. Applications of graphs in computer science.	Rosen: Chapter 5

Course Title	**Functional English**		
Course Code	**GE-162**		
Credit Hours	3 (3,0)		
Category	General Education		
Prerequisite	None		
Co-Requisite	None		
Follow Up	Expository Writing		
Course Introduction	This course is designed to equip students with essential language skills for effective communication in diverse real-world scenarios. It focuses on developing proficiency in English language usage: word choices, grammar and sentence structure. In addition, the course will enable students to grasp nuanced messages and tailor their communication effectively through application of comprehension and analytical skills in listening and reading. Moreover, the course encompasses a range of practical communication aspects including professional writing, public speaking, and everyday conversation, ensuring that students are equipped for both academic and professional spheres. An integral part of the course is fostering a deeper understanding of the impact of language on diverse audiences. Students will learn to communicate inclusively and display a strong commitment to cultural awareness in their language use. Additionally, the course will enable them to navigate the globalized world with ease and efficacy, making a positive impact in their functional interactions.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Apply enhanced English communication skills through effective use of word choices, grammar, and sentence structure	C4 (Apply)	1,6,7,10
	CL2: Comprehend a variety of literary / non-literary written and spoken texts in English.	C2 (Understand)	1,6,7,10
	CLO3: Effectively express information, ideas and opinions in written and spoken English	C2 (Explain)	1,6,7,10
	CLO4: Recognize inter-cultural variations in the use of English language and to effectively adapt their communication style and content based on diverse cultural and social contexts.	C4 (Identify)	1,6,7,10
Text Book(s)	1. "The Blue Book of Grammar and Punctuation" by Jane Straus. 2. "Practical English Usage" by Michael Swan 3. "Business Communication Today" by Courtland L.Bovee		
Reference Material	1. "Understanding and Using English Grammar" by Betty Schrampfer Azar. 2. "English Grammar in Use" by Raymond Murphy. 3. "English for Specific Purposes: A Learning-Centered Approach" by Tom Hutchinson and Alan Waters. 4. "Cambridge English for Job-hunting" by Colm Downes. 5. "Reading Literature and Writing Argument" by Missy James and Alan P. Merickel. 6. "Improving Reading: Strategies, Resources, and Common Core Connections" by Jerry Johns and Susan Lenski. 7. "Comprehension: A Paradigm for Cognition" by Walter Kintsch. 8. "Communication Skills for Business Professionals" by J.P. Verma and Meenakshi Raman.		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Vocabulary building (contextual usage, synonyms, antonyms and idiomatic expressions)	Langan: Chapter 23,26,27,28

Week 2	Subject-verb-agreement, Verb tenses	Langan: Chapter 23,26,27,28
Week 3	Fragments, Run-ons, Modifiers	Langan: Chapter 24,25,32,33 Quiz 1
Week 4	Articles	Murphy: Chapter 69-81
Week 5	Word classes	Swan: Grammar section 1-28
Week 6	Word formation (affixation, compounding, clipping, back formation, etc).	Swan: Chapter 30 Quiz 2
Week 7	Sentence structure (simple, compound, complex and compound-complex), Sound production and pronunciation.	Swan: Chapter 27
Week 8	Comprehension and Analysis: Understanding purpose, audience and context, Contextual interpretation (tones, biases, stereotypes, assumptions, inferences, etc.)	Zoe L. Albright &Langan: Part 7 Quiz 3
Week 9	Reading strategies (skimming, scanning, SQ4R, critical reading, etc.), Active listening (overcoming listening barriers, focused listening, etc.)	Bovee: Chapter 2
Week 10	Effective Communication: Principles of communication (clarity, coherence, conciseness, courteousness, correctness, etc.)	Quiz 4
Week 11	Structuring documents (introduction, body, conclusion and formatting), Inclusivity in communication (gender-neutral language, stereotypes, cross-cultural communication, etc.	Bovee: Chapter 3
Week 12	Public speaking (overcoming stage fright, voice modulation and body language)	Swan: Chapter 27 Quiz 5
Week 13	Presentation skills (organization content, visual aids and engaging the audience)	Bovee: Chapter 16
Week 14	Presentation skills (structured and free-form slides, linear and non-linear presentations)	Bovee: Chapter 17 Presentation
Week 15	Professional writing (business e-mails, memos, business letter)	Bovee: Appendix-A
Week 16	Professional writing (Reports and proposals)	Bovee: Chapter 13,14,15 Assignment

Course Title	**Introduction to Economics**		
Course Code	**GE-163**		
Credit Hours	2 (2,0)		
Category	General Education (Social Science)		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course introduces the fundamental principles of economics, focusing on how individuals, firms, and governments allocate scarce resources. It covers microeconomic and macroeconomic concepts, including consumer behavior, market mechanisms, production and cost analysis, national income, inflation, and fiscal and monetary policies. Emphasis is placed on analytical thinking and real-world applications to understand economic issues at national and global levels.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Explain fundamental economic concepts, including demand, supply, and national income.	C2 (Understand)	1,3
	CLO2: Apply economic principles to analyze market behavior and real-world issues.	C3 (Apply)	1,2,3
	CLO3: Analyze government policies and global economic interactions.	C4 (Analyze)	1,3,8
Text Book(s)	Economics 23rd Edition by Campbell R. McConnell, Stanley L. Brue and Sean Masaki Flynn McGraw Hill Education, 2023. ISBN: 1266106847, 978-1266106842		
Reference Material	Economics Today 20th Edition by Roger LeRoy Miller., Pearson, 2011. ISBN 0135857309, 978-0135857304		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to economics. Nature and scope of economics. Subject matter of economics. The circular flow of income and product. Basic economic problems and scarcity. Role of government in economic systems.	McConnell: Chapter 1
Week 2	Society's technological possibilities and production possibilities frontier (PPF). Opportunity cost and economic efficiency. Introduction to microeconomics and macroeconomics.	McConnell: Chapter 2
Week 3	Theory of consumer behavior. Preferences, utility, and rational decision-making. Basic analysis of consumer choice.	McConnell: Chapter 3 Quiz1
Week 4	Theory of demand. Law of demand. Determinants of demand. Demand schedule and demand curve. Shifts in demand.	McConnell: Chapter 4 Quiz2
Week 5	Theory of supply. Law of supply. Determinants of supply. Supply schedule and supply curve. Market equilibrium.	McConnell: Chapter 5
Week 6	Determination of value (price) of a commodity. Interaction of demand and supply. Market mechanism and price determination.	McConnell: Chapter 6 Quiz3
Week 7	Elasticity of demand. Types and measurement of elasticity. Applications of elasticity in real-world pricing decisions.	McConnell: Chapter 7 Quiz4
Week 8	Elasticity of supply. Measurement and interpretation. Mid-term review of market concepts.	McConnell: Chapter 8
Week 9	Types of markets. Perfect competition, monopoly, monopolistic competition, and oligopoly. Characteristics and comparison.	McConnell: Chapter 9

Week 10	Revenue concepts. Total revenue, average revenue, and marginal revenue. Relationship between revenue curves under different market structures.	McConnell: Chapter 10 Quiz5
Week 11	Cost analysis. Fixed and variable costs. Average and marginal costs. Production cost, input cost, overhead cost, and marketing cost.	McConnell: Chapter 11
Week 12	Industry analysis with focus on the software industry. Factor markets and pricing of factors of production.	McConnell: Chapter 12 Quiz6
Week 13	National income accounting. Measurement of GDP and national income. Economic growth and standard of living.	McConnell: Chapter 13
Week 14	Macroeconomic concepts. Aggregate demand and aggregate supply. Inflation and unemployment. Role of taxation and subsidies.	McConnell: Chapter 14 Quiz7
Week 15	Role of government in the economy. Fiscal policy and federal budget. Central bank and monetary policy. Open economy concepts including money and finance.	McConnell: Chapter 15 Quiz8
Week 16	Global economics. Free trade and protectionism. Role of international organizations (WTO). Trade development and economic globalization. Types of states (welfare and non-welfare).	McConnell: Chapter 16

Course Title	**Probability & Statistics**		
Course Code	**GE-164**		
Credit Hours	3 (3,0)		
Category	General Education (Quantitative Reasoning – II)		
Prerequisites	None		
Co-Requisite	None		
Follow Up	CSE-322 Machine Learning		
Course Introduction	This course introduces the fundamental principles of probability theory and various descriptive statistical techniques for collecting, analyzing, and interpreting data. The course also covers inferential statistics, which includes sampling, estimation of parameters, and testing of hypotheses. Main topics include variables and their types, patterns of variations, randomness of variables, combinations, probability distributions, Bayesian inference, and linear regression.		
Course Learning	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Describe fundamental principles of probability theory, data types, and descriptive statistical techniques.	C2 (Understand)	1,2
	CLO2: Apply probability distributions and sampling techniques to solve computing and engineering problems	C3 (Apply)	2,3
	CLO3: Analyze datasets through hypothesis testing and regression modeling to reach substantiated conclusions.	C4 (Analyze)	3,4
Text Book(s)	1. Probability & Statistics for Engineers & Scientists by Ronald Walpole, Raymond Myers, et al. (9th Edition, 2016)		
Reference Material	1. David Forsyth (2017), Probability and Statistics for Computer Science. 2. Prof. Sher Muhammad Chaudhry & Prof. Dr. Shahid Kamal (2014), Introduction to Statistical Theory, Part-1 & part 2, Ilmi Kitab Khana. 3. Sheldon M. Ross (2020). Probability and statistics for engineers and scientists (6th Ed.). Elsevier.		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to Statistics, Descriptive and Inferential Statistics. Population vs. Sample; Data, and their types as Qualitative and Quantitative, Discrete/Continuous. Scales of measurement: Nominal, Ordinal, Interval, Ratio.	Walpole: Chapter 1 (Sec. 1.1—Sec. 1.3)
Week 2	Graphical Representation of Data: Histogram, Bar chart, Summary Tables, Pie Charts. Dot-plots/Scatter diagram, Stem-and Leaf Plot, Histogram,	Walpole: Chapter 1 (Sec. 1.4—Sec. 1.5) Assignment 1
Week 3	Frequency/Relative/Percentage Frequency distributions, Cumulative Frequencies, CF Curves.	Walpole: Chapter 1 (Sec. 1.6)
Week 4	Measures of central tendency: Arithmetic Mean, Geometric Mean, Harmonic Mean, Median, Mode.	Walpole: Chapter 1 (Sec. 1.7) Quiz 1
Week 5	Measures of Dispersion and their properties: Range, Quartiles and Boxplot, Quartile Deviation, mean deviation, percentiles. Describing Data using Chebysheve Rule,	Walpole: Chapter 1 (Sec. 1.8) Walpole:, Chapter 4 (Sec. 4.4) Assignment 2

Week 6	Empirical Rule, Variance, Standard Deviation, Skewness & Kurtosis. Introduction to probability theory, random experiment, sample space, simple and compound events. Assigning probabilities. Prior-Classical method, empirical method, subjective method.	Walpole: Chapter 1 (Sec. 1.9) Walpole: Chapter 2 (Sec. 2.1—Sec. 2.4) Quiz 2
Week 7	Additive Rule. Venn Diagram Examples, Compound Events and their probabilities. Multiplication Rule of probabilities.	Walpole: Chapter 2 (Sec. 2.5—Sec. 2.6) Quiz 3
Week 8	Conditional Probability, Independent Events.	Walpole: Chapter 2 (Sec. 2.6)
Week 9	Law of total probability and Bayes Rule. Examples and Exercises on prior and posterior probabilities.	Walpole: Chapter 2 (Sec. 2.7)
Week 10	Introduction to Random Variable, Discrete Random Variables and their Probability Distributions, Expected value and Variance.	Walpole: Chapter 3 (Sec. 3.1—Sec. 3.2) Walpole: Chapter 4 (Sec. 4.1—Sec. 4.2) Assignment 3
Week 11	Binomial Random Variable and its Probability Distribution, Mean and Variance of Binomial random variable. Poisson Random Variable its Distribution Function and Probabilities. Poisson Approximation to Binomial Distribution	Walpole: Chapter 5 (Sec. 5.1—Sec. 5.2 & Sec. 5.5) Quiz 4
Week 12	Introduction to Continuous Random Variables and their Probabilities as definite integrals. Uniform distribution. Introduction to Normal Distribution.	Walpole: Chapter 3 (Sec. 3.3) Walpole: Chapter 6 (Sec. 6.1—Sec. 6.3) Presentation Assigned.
Week 13	Examples on finding probabilities of a normally distributed random variable by using Standard Normal Curve. Normal Approximation to Binomial Distributions.	Walpole: Chapter 6 (Sec. 6.4—Sec. 6.5) Quiz 5
Week 14	Statistical Inference I: Point Estimation, Hypothesis Testing: Null/Alt Hypotheses, Type I & II Errors, P-values.	Walpole: Chapter 9 (Sec. 9.1—Sec. 9.2) Walpole: Chapter 10 (Sec. 10.1—Sec. 10.3)
Week 15	Linear Regression I: Least Squares Method, Mean Square Error (MSE) in Regression.	Walpole: Chapter 11 (Sec. 11.1—Sec. 11.3)
Week 16	Final Presentations	Final Presentations

Course Title	**Expository Writing**		
Course Code	**GE-165**		
Credit Hours	3 (3,0)		
Category	General Education		
Prerequisite	Functional English		
Co-Requisite	None		
Follow Up	None		
Course Introduction	Expository Writing is a sequential undergraduate course aimed at refining writing skills in various contexts. Building upon the foundation of the pre-requisite course, Functional English, this course will enhance students' abilities of producing clear, concise and coherent written texts in English. The course will also enable students to dissect intricate ideas, to amalgamate information and to express their views and opinions through well-organized essays. The students will further be able to refine their analytical skills to substantiate their viewpoints using credible sources while adhering to established ethical writing norms. Additionally, the course will highlight the significance of critical thinking enabling students to produce original and engaging written texts.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Understand the essentials of the writing process integrating pre-writing, drafting, editing and proof reading to produce well-structured essays.	C2 (Understand)	1,6,7,10
	CLO2: Demonstrate mastery of diverse expository types to address different purposes and audiences.	C3 (Apply)	1,6,7,10
	CLO3: Uphold ethical practices to maintain originality in expository writing.	C3 (Demonstrate)	1,6,7,10
Text Book(s)	1. "College Writing Skills" by John Langan		
Reference Material	1. https://owl.purdue.edu/ (For Citation Styles) 2. "Exploring Writing: Paragraphs and Essays" by Zoe L. Albright and John Langan		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Understanding expository writing (definition, types, purpose and applications), Characteristics of effective expository writing (clarity, coherence and organization)	Zoe L. Albright and John Langan: Chapter 1 &2
Week 2	Introduction to paragraph writing.	Zoe L. Albright and John Langan: Chapter 1 &2
Week 3	The Writing Process: Pre-writing techniques (brainstorming, free-writing, mind-mapping, listing, questioning and outlining etc.), Drafting (three stage process of drafting techniques)	Langan: Chapter 2 Quiz 1
Week 4	Revising and editing (ensuring correct grammar, clarity, coherence, conciseness etc.), Proof reading (fine-tuning of the draft), Peer review and feedback (providing and receiving critique).	Langan: Chapter 2 & 5
Week 5	Essay Organization and Structure: Introduction and hook (engaging readers and introducing the topic), Thesis statement (crafting a clear and focused central idea)	Langan: Chapter 3 Quiz 2
Week 6	Essay Organization and Structure: Body Paragraphs (topic sentences, supporting evidence and transitional devices), Conclusion (types of concluding paragraphs and leaving an impact), Ensuring cohesion and coherence (creating seamless connections between paragraphs).	Langan: Chapter 4,5,6

Week 7	Different Types of Expository Writing: Exemplification, Narration	Langan: Chapter 9 &10 Quiz 3
Week 8	Different Types of Expository Writing: Description, Process, Argument Writing	Langan: Chapter 8, 11 &16
Week 9	Different Types of Expository Writing: Cause and Effect, Compare and/or Contrast	Langan: Chapter 12 & 13 Quiz 4
Week 10	Different Types of Expository Writing: Definition, Division-Classification	Langan: Chapter 14 & 15
Week 11	Writing for Specific Purposes and Audiences: Different types of purposes (to inform, to analyze, to persuade, to entertain etc.)	Bovee: Chapter 4 Quiz 5
Week 12	Writing for academic audiences (formality, objectivity, and academic conventions), Writing for public audiences (engaging, informative and persuasive language), Different tones and styles for specific purposes and audiences.	Bovee: Chapter 5
Week 13	Ethical Considerations: Ensuring original writing (finding credible sources, evaluating information etc.)	Bovee: Chapter 4
Week 14	Ethical Considerations: Proper citation and referencing (APA and MLA)	https://owl.purdue.edu/ Quiz 6
Week 15	Ethical Considerations: Proper citation and referencing (IEEE)	https://owl.purdue.edu/
Week 16	Integrating quotes and evidences (quoting, paraphrasing, and summarizing), Avoiding plagiarism (ethical considerations and best practices)	Bovee: Chapter 13 Quiz 7

Course Title	**Professional Practices**		
Course Code	**GE-263**		
Credit Hours	2 (2,0)		
Category	General Education (Arts and Humanities)		
Prerequisite	None		
Co-Requisite	None		
Follow-up	None		
Course Introduction	A computing graduate as a professional has some responsibilities with respect to society. This course develops students' understanding of historical, social, economic, ethical, and professional issues related to the discipline of computing. It identifies key sources for information and opinions about professionalism and ethics. Students analyze, evaluate, and assess ethical and professional computing case studies.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Trace the historical evolution of the computing profession and its impact on society.	C1 (Know)	1,8
	CLO2: Describe the interplay between computing technologies and societal shifts, acknowledging both positive and negative implications.	C2 (Describe)	1,8,9,10
	CLO3: Recognize and explain the core ethical principles that guide the computing profession.	C4 (Identify)	1,8,9,10
	CLO4: Explain the responsibilities of computing professionals in their interactions with society and individuals.	C2 (Explain)	1,8,9,10
	CLO5: Analyze and critically evaluate real-world case studies in computing, assessing them from both ethical and professional viewpoints.	C4 (Analyze)	1,8,9,10
Text Book(s)	1. Michael J. Quinn, Ethics for the Information Age, 7th Edition, Pearson Education, 2017, ISBN: 978-0134296548 2. Sara Baase, A Gift of Fire: Social, Legal, and Ethical Issues in Computing, 5th Edition, Pearson, 2018, ISBN: 978-0132492676		
Reference Material	1. J. Kizza, Ethical and Social Issues in the Information Age, 6th Edition, Springer, 2017, ISBN: 978-3319707112 2. "Professional Issues in Software Engineering" by Frank Bott, Allison Coleman, Jack Eaton and Diane Rowland, 3rd Edition, CRC Press, 2000. ISBN-10: 0748409513 3. Computer Ethics by Deborah G. Johnson, Pearson; 4th Edition, 2009. ISBN-10: 0131112414		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	The definition of Student, Learning, DE learning, relearning, Zero-line theory Data, information, knowledge, skills, experience Foundation of Professional Practices	Handouts
Week 2	What is Ethics? The importance of integrity. The difference between morals, ethics, and law. Ethics in the business world. Why Fostering Good Business Ethics Is Important.	Baase: Section 1.4.1 Assignment 1 Quiz 1
Week 3	The tool kit for ethics, manners, and etiquettes: Management and leadership	Quinn et al.: Section 2.7, 2.6 Baase: Section 1.4.2
Week 4	The tool kit for ethics, manners, and etiquettes	Quinn et al.: Section 2.7, 2.6

		Baase: Section 1.4.2
Week 5	The tool kit for ethics, manners, and etiquettes: Anger and stress Management	Quinn et al.: Section 2.7, 2.6 Baase: Section 1.4.2 Assignment 2 Quiz 2
Week 6	Professional Ethics: Introduction: Why Professional Ethics?, The Paradigm of Professions, Characteristics of a fully developed Profession, Sorting Out Computing and its Status as a Profession, Common morality, Personal ethics, Professional ethics	Baase: Section 1.4.2 Quinn et al.:2.9.1, 2.10 Baase: Section 1.4.3
Week 7	Professional Ethics: Communication (Writing, speaking, Listening)	Handouts Assignment 3 Quiz 3
Week 8	Professional Ethics: How to criticize, How to ask question, How to interrupt	Handouts Assignment 4 Quiz 4
Week 9	Computer Ethics: Technology and Ethics, Computer ethics and morality, Ethical decision making in computing, The special status of computer ethics (uniquely stored, uniquely malleable, uniquely complex, uniquely fast, uniquely cheap, uniquely cloned, uniquely discrete, uniquely coded) with scenarios	Quinn et al.: Section 9.1, 9.2 Kizza: Section 2
Week 10	Ethical and Social Impacts of Computers on Work: Changes, fears and questions, Impacts on employment (Job destruction and creation, changing skills and skill levels, telecommuting, a global workforce), Employee communication and monitoring (Learning about job applicants, risks and rules for work and personal communications)	Quinn et al.: Section 10.1, 10.2, 10.3 Baase: Section 6.1, 6.2 Baase: Section 6.3
Week 11	Ethical Evaluation of Technology: Evaluating information (the need for responsible judgment), The "Digital Divide" (Trends in computer access, the global divide and next billion users), Neo-Luddite views of computers, technology and quality of life (Criticism of computing technologies), Making decision about technology (Intelligent machines and super-intelligent humans – or the end of human race?) Ethical issues in design process	Quinn et al.: Section 10.4, 10.5 Baase: Section 7.2 Baase: Section 7.3, 7.4 Assignment 5 Quiz 5
Week 12	Ethics and the Internet: Three morally significant characteristics (global, anonymity, reproducibility), ethical significance Hacking and Hackers' Ethics: What is hacking? Hacktivisim and political hacking, Hackers as security researchers, Hacking as foreign policy, Security, The law: catching and punishing hackers	Baase: Section 5.2 Quinn et al.: Section 7.2, 7.4 Assignment 6 Quiz 6
Week 13	Identity Theft: Identity theft and credit card fraud, stealing identities, Responses to identity theft, biometrics, whose laws rule the web? When digital actions cross border, culture, law and ethics, potential solutions Case study	Quinn et al.: Section 8.1, 8.2, 8.3, 8.6, 8.7 Assignment 7 Quiz 7
Week 14	Ethics and Privacy: What is privacy?, New technology, new risks, understanding the "Computers and Privacy" issues (uses of information, personal privacy, information mediates relationships, individual-organization relationships), Exception of privacy and surveillance	Quinn et al.: Section 5.1, 5.2 Baase: Section 2.1, 2.3, 2.4 Assignment 8

	technologies, The business and social sectors, Government systems, Protecting Privacy: Technology, Markets, Rights, and Laws	
Week 15	Ethics and Intellectual Property: What is intellectual property? Challenges of new technology, Current legal protection, The philosophical basis of property, Consequentialist arguments, Conclusions from the philosophical analysis of property, Is it wrong to copy proprietary software? Scenarios	Quinn et al.: Section 4.1, 4.2, 4.3, 4.7, 4.8, 4.9 Baase: Section 4.1 Quiz 8
Week 16	The Software Engineering Code of Ethics: Professional Relationships (Employer-employee, Client-professional, Other stakeholders-professionals, Professional-professional, Conflicting responsibilities) ACM Code of Ethics and Professional Conduct, IEEE Code of Ethics, DPMA Code of Ethics, ICCP Code of Ethics	Baase: Appendix A

Course Title	**Pakistan Studies**		
Course Code	GE-166		
Credit Hours	2 (2,0)		
Category	General Education		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course is designed to provide students with a comprehensive exploration of Pakistan's identity, spanning geographical, historical, and cultural dimensions. It delves into the diverse landscapes, ancient civilizations, and rich cultural heritage that define Pakistan. Moreover, it examines the socio-cultural and political transformations in Pakistan over time, including democratic transitions and military interventions. The aim of this course is to inculcate in students a nuanced understanding of Pakistan's past, present, and potential future trajectories, enabling them to critically evaluate the complex dynamics shaping the nation's development.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Have enhanced knowledge of the geographical, historical, and political aspects of Pakistan	C1 (Explain)	1,2
	CLO2: Understand the society and culture of Pakistan	C3 (Identify)	1,2
	CLO3: Understand and explain the socio-economic developments in Pakistan.	C3 (Determine)	2,3
	CLO4: Explore contemporary issues and challenges faced by Pakistan and their implications for the future.	C1 (Know)	1.2
Text Book(s)	Muhammad Raza Kazmi, *Pakistan Studies* (Karachi: Oxford University Press, 2007).		
Reference Material	"Trek to Pakistan" by Ahmad Saeed and K.K. Mansur Sarwar "Pakistan: A Modern History" by Ian Talbot		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to Pakistan Studies: Definition, scope and significance of Pakistan Studies.	Hayat: Chapter 1
Week 2	Geographical location and importance of Pakistan.	Hayat: Chapter 3 Quiz 1
Week 3	Historical background: Ancient civilizations (Indus Valley Civilization).	Hayat: Chapter 2 Assignment 1
Week 4	Freedom movement and factors leading to the creation of Pakistan.	Hayat: Chapter 4 Quiz 2
Week 5	Political History of Pakistan: Early challenges and formative phase (1947–1958).	Hayat: Chapter 4
Week 6	Military interventions and democratic transitions.	Hayat: Chapter 4 Quiz 3
Week 7	Geography of Pakistan: Physical features (mountains, plains, plateaus, deserts).	Hayat: Chapter 3 Assignment 2
Week 8	Climate and environmental regions of Pakistan	Hayat: Chapter 3
Week 9	Indus River system and its significance.	Hayat: Chapter 3 Presentation
Week 10	Society and Culture: Cultural diversity and social structure.	Hayat: Chapter 6 Quiz 5

Week 11	Languages and literature of Pakistan.	Hayat: Chapter 9 Presentation
Week 12	Economic Development:Agriculture and industrial sectors.	Hayat: Chapter 5 Quiz 6
Week 13	Economic challenges (inflation, unemployment, debt).	Hayat: Chapter 5 Presentation
Week 14	Contemporary Issues: Terrorism, extremism, and regional conflicts.	Recent Literature Quiz 7
Week 15	Environmental issues and sustainable development	Recent Literature
Week 16	Media and social change in Pakistan.	Revision

Course Title	**Fehm-e-Quran I**		
Course Code	**GE-167**		
Credit Hours	1 (0,3)		
Category	General Education		
Prerequisite	None		
Co-Requisite	None		
Follow Up	Fehm-e-Quran II		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Understanding the meanings of basic words of the Quran, phrases and sentences.	C1 (Remember)	2
	CLO2: Recognize and understand different styles of Quranic sentences.	C2 (Understand)	2,8
	CLO3: Comprehend Quranic vocabulary, particles (operative & non-operative particles), compounds (Adjective & Possessive compound), pronouns (singular & plural), and types of plural through hundreds of Quranic sentences.	C3 (Apply)	2,10
Course Description	This course is designed to develop the ability to understand basic words of the Quran, phrases and sentences that do not contain verbs and sentences having present tense. Acquire a strong foundation for understanding long verses of the Quran with clarity. Comprehend Quranic vocabulary, particles (operative & non operative particles), compounds (Adjective & Possessive compound), pronouns (singular & plural) and types of plural through hundreds of Quranic sentences. Recognize and understand different styles of Quranic sentences, including nominal sentence, emphatic sentence, double emphatic sentence, negative sentence, interrogative sentence, oath –based sentences.		
Text Book(s)	Muallimul Quran by Dr. Ubaidurrahman volume 1,2,3		
Reference Material	The Holy Quran Translations.		

Weekly Lesson Plan		
Week No.	**Course Content**	**Assignments/Readings**

Weeks	Lectures (1.5 hrs)	Units	Lessons	Assignments/Home Task	Linguistic Rules
1.	1.	1	1-6	Writing the meaning of Quranic words Lesson 1-8	Proper Noun Masculine & Feminine
	2.	1	9-14	Writing the meaning of Quranic words 9-14	Two kinds of plural Concept of (و) "And" Common Noun
2.	1.	1	15-17	Writing the meaning of Quranic words, phrases & translation of Sentences 15-17	Demonstrative Noun (This & That for Masculine (هذا- هذه) Demonstrative Noun (This & That for Feminine) (ذلك- تلك)
	2.	1	18-19 & Revision (Unit 1)	Writing the meaning of Quranic words , phrases & translation of Sentences 17-19 Quiz	Laam for emphasis (لام التأكيد) Superlative Degree like أكبر Revision of all Quranic Sentences
3.	1.	Unit 2	1-3	Writing the meaning of Quranic words, phrases & translation of Sentences 1-3	Emphatic Particle إن Preposition "For" (اللام) Preposition (في)
	2.	2	4-6	Writing the meaning of Quranic words, phrases & translation of Sentences 4-6	Preposition (على- من- إلى)
4.	1.	2	7- 9	Writing the meaning of Quranic words & translation of Sentences 7-9	Preposition (الباء) Absolute Negation Particle Exceptive Particle (لا النافية للجنس)(ما النافية) (إلا)
	2.	2	10-13 & Revision (Unit 2)	Writing the meaning of Quranic words, phrases & translation of Sentences 10-13 Quiz	Subordinating Conjunction(أن), Was (كان), Vocative Particle(حرف النداء)

5.	1.	Unit 3	1-2	Writing the meaning of Quranic phrases 1-2	Quranic Adjective Compounds (صفة وموصوف)
	2.	3	3-5	Writing the meaning of Quranic phrases & translation of sentences 3-5	Quranic Possessive Construction (مضاف ومضاف إليه)
6.	1.	3	6 -7	Writing the meaning of Quranic phrase translation of sentences 6-7	Quranic Possessive Construction (مضاف ومضاف إليه)
	2.	3	8-10 & Revision (Unit 3)	Writing the meaning of Quranic phrase & translation of sentences 8-10 Quiz	Active Participle (اسم الفاعل), Passive Participle (اسم المفعول), Dual (مثنى)
7.	1.	Unit 4	1-2	Writing the meaning of Quranic phrase & translation of sentences 1-2	Personal Pronoun He(هو المنفصل) Possessive Pronoun His ـه المتصل)
	2.	4	3-4	Writing the meaning of Quranic phrase & translation of sentences 3 -4	Possessive Pronoun with prepositions like في بيته Pronoun "His" with prepositions like له، منه، فيه
8.	1.	4	5-8	Writing the meaning of Quranic sentences 5-8	Personal Pronoun You(أنت المنفصل) Possessive Pronoun Your ك المتصل) Possessive Pronoun with prepositions like في بيتك Pronoun "your" with prepositions like لك، منك، فيك
	2.	Mid Term			

9.	1.	4	9-12	Writing the meaning of Quranic phrases & sentences 9-12	Personal Pronoun She(هي المنفصل) Possessive Pronoun Her ها المتصل) Possessive Pronoun with prepositions like في بيتها Pronoun "Her" with prepositions like لها،
	2.	4	13-16	Writing the meaning of Quranic phrases & sentences 13-16	Personal Pronoun I (أنا المنفصل) Possessive Pronoun Her ي المتصل) Possessive Pronoun with prepositions like في بيتيPronoun "My" with prepositions like لي
10.	1	4	17 & Revision Unit 4	Revision of all Quranic sentences of Unit 4 Quiz	Adverb (حال)
	2.	Unit 5	1-2	Writing the meaning of Quranic phrases & sentences 1-2	Masculine Plural جمع المذكر السالم و جمع المذكر السالم المسبوق بحرف الجر
11.	1.	5	3-4	Writing the meaning of Quranic phrases & sentences 3-4	Possessive Construction with Plurals جمع المذكر السالم المسبوق بالإضافة
	2.	5	5-6	Writing the meaning of Quranic phrases, sentences & verses 5-6	Personal Pronoun They (هم المنفصل) Possessive Pronoun Their هم المتصل)
12.	1.	5	7-8	Writing the meaning of Quranic phrases, sentences & verses 7-8	Possessive Pronoun with prepositions like في بيتهمPronoun "Their" with prepositions like لهم
	2.	5	9-11	Writing the meaning of Quranic phrases, sentences & verses 9-11	Personal Pronoun You (أنتم المنفصل) Possessive Pronoun Your كم المتصل) Possessive Pronoun with prepositions

					like في بيتكم
13.	1.	5	12-14	Writing the meaning of Quranic phrases & sentences & verses 12-14	Pronoun "Your" with prepositions like لكم Personal Pronoun We (نحن المنفصل) Possessive Pronoun Our نا المتصل)
	2.	5	15-16	Writing the meaning of Quranic sentences & verses 15-16	Possessive Pronoun with prepositions like في بيتنا Pronoun "Our" with prepositions like لنا
14.	1.	5	17-18	Writing the meaning of Quranic sentences & Verses 17-18	Demonstrative Pronoun These, Those (هؤلاء- أولئك)
	2.	5	19-23	Writing the meaning of Quranic sentences & Verses 19-23	ما / إلا، إن / إلا، إنما، ليس، ما ،) (أ/أم، أن، بل، كأن (ألا، أليس، اليوم، يومئذ، سبحان، مابينهما، قل، إذن، بئس، نعم، كلا، ما أدراك، حسب، أعلم ب، مصير، مرجع، دينا(تمييز)
15.	1.	5	Revision Unit 5	Quiz	
	2.	5	1-3 (till Page 16)	Writing the meaning of Quranic Verbs &Translation of Quranic Sentences & Verses (1-3)	Introduction of Present Tense(فعل مضارع) & Verbal Sentence (جملة فعلية) Present Tense الفعل المضارع صيغة المفرد يعلم
16.	1.	6	3 (From Page 17) & 4 -5	Translation of Quranic Sentences & Verses 3-5	Present Tense الفعل المضارع صيغة المفرد يعلم
	2.	6	6	Translation of Quranic Sentences & Verses	Present Tense الفعل المضارع صيغة الجمع يعلمون

Course Title	**Fehm-e-Quran II**		
Course Code	**GE-168**		
Credit Hours	1 (0,3)		
Category	General Education		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Understanding the meanings of basic words of the Quran, phrases and sentences.	C1 (Remember)	2
	CLO2: Recognize and understand different styles of Quranic sentences.	C2 (Understand)	2,8
	CLO3: Comprehend Quranic vocabulary, particles (operative & non-operative particles), compounds (Adjective & Possessive compound), pronouns (singular & plural), and types of plural through hundreds of Quranic sentences.	C3 (Apply)	2,10
Course Description	This course is designed to develop the ability to understand basic words of the Quran, phrases and sentences that do not contain verbs and sentences having present tense. Acquire a strong foundation for understanding long verses of the Quran with clarity. Comprehend Quranic vocabulary, particles (operative & non operative particles), compounds (Adjective & Possessive compound), pronouns (singular & plural) and types of plural through hundreds of Quranic sentences. Recognize and understand different styles of Quranic sentences, including nominal sentence, emphatic sentence, double emphatic sentence, negative sentence, interrogative sentence, oath –based sentences.		
Text Book(s)	Muallimul Quran by Dr. Ubaidurrahman volume 1,2,3		
Reference Material	The Holy Quran Translations.		

Weekly Lesson Plan

Weeks	Lectures	Units	Lessons	Assignments/Home Task	
1.	1.	6	6	Understanding & Translation of Verses	Present Tense صیغة جمع مذکر غائب مثل یعبدون
	2.	6	7-8	Understanding & Translation of Verses	Present Tense صیغة جمع مذکر غائب مثل یعبدون
2.	1.	6	9-10	Understanding & Translation of Verses	Present Tense صیغة مفرد مذکر مخاطب (تعبد) وجمع مذکر مخاطب (تعبدون)
	2.	6	11-12	Understanding & Translation of Verses	Present Tense صیغة جمع مذکر مخاطب (تعبدون)

					صيغة المتكلم (أعبد)
3.	1.	6	13	Understanding & Translation of Verses	Present Tense صيغة جمع المتكلم (نعبد)
	2.	6	14-15	Understanding & Translation of Verses	Negative Imperative صيغة المفرد وصيغة الجمع , لا تعبدْ، لا تعبدوا
4.	1.	6	16-17	Understanding & Translation of Verses	Conditional Sentences & masdar moawal (مصدر موؤل)
	2.	6	18-19	Understanding & Translation of Verses	Laam uttaleel (لام التعليل) & Laam ul jhood(لام الجحود)
5.	1.	6	20-21	Understanding & Translation of Verses	Present with object pronouns & Passive Voice
	2.	6	Revision (Unit 6)	Quiz	
6.	1.	Unit 7	1 (sec 1-3)	Understanding & Translation of Verses	Past Tense صيغة المفرد الغائب
	2.	6	1 (Sec 4-5)	Understanding & Translation of Verses	Past Tense صيغة المفرد للغائب
7.	1.	6	1 (Sec 5-6)	Understanding & Translation of Verses	Past Tense صيغة المفرد للغائب
	2.	6	1 (Sec 7-9)	Understanding & Translation of Verses	Past Tense صيغة المفرد للغائب
8.	1.	7	Revision	Understanding & Translation of Verses QUIZ	Past Tense صيغة المفرد للغائب
	2.	MID TERM			
9.	1.	7	2 (sec 1-2)	Understanding & Translation of Verses	Past Tense صيغة الجمع للغائب عبدوا
	2.	7	2 (sec 3)	Understanding & Translation of Verses	Past Tense صيغة الجمع للغائب عبدوا
10.	1.	7	2 (sec 4-5)	Understanding & Translation of Verses	Past Tense صيغة الجمع للغائب عبدوا
	2.	7	2 (sec 6-7)	Understanding & Translation of Verses	Past Tense صيغة الجمع للغائب عبدوا
11.	1.	7	3 (sec 1-2)	Understanding & Translation of Verses	Past Tense صيغة الجمع للمتكلم عبدنا

	2.	7	3 (sec 2-3)	Understanding & Translation of Verses	Past Tense صيغة الجمع للمتكلم عبدنا
12.	1.	7	3 (sec 3-4)	Understanding & Translation of Verses	Past Tense صيغة الجمع للمتكلم عبدنا
	2.	7	3 (sec 4-5)	Understanding & Translation of Verses	Past Tense صيغة الجمع للمتكلم عبدنا
13.	1.	7	4 (sec 1-2-3)	Understanding & Translation of Verses	Past Tense صيغة الجمع للمخاطب عبدتم
	2.	7	4 (sec 4-5)	Understanding & Translation of Verses	Past Tense صيغة الجمع للمخاطب عبدتم
14.	1.	7	5-6	Understanding & Translation of Verses Quiz	Past Tense صيغة المتكلم والمخاطب عبدتُ ، عبدتَ
	2.	7	7	Understanding & Translation of Verses	Past Tense صيغة المؤنث للغائب عبدَتْ
15.	1.	7	8	Understanding & Translation of Verses	Passive Voice (Past Tense) فعل مجهول للمفرد
	2.	7	9	Understanding & Translation of Verses	Passive Voice (Past Tense) فعل مجهول الجمع
16.	1.	8	1-4	Understanding & Translation of Verses	Imperative Verb for singular فعل الأمر للمفرد
	2.	7	5-8	Understanding & Translation of Verses	Imperative Verb for plural فعل الأمر للجمع

Course Title	**Civics and Community Management**		
Course Code	**GE-260**		
Credit Hours	2 (2,0)		
Category	General Education		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course is designed to provide students with fundamental knowledge about civics, citizenship, and community engagement. Students will learn about the essentials of civil society, government, civic responsibilities, inclusivity, and effective ways to participate in shaping society, which will help them apply theoretical knowledge to real-world situations to make a positive impact on their communities.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Demonstrate fundamental understanding of civics, government, citizenship and civil society.	C2 (Understand)	1
	CLO2: Understand the concept of community and recognize the significance of community engagement for individuals and groups.	C2 (Understand)	1,3,6,7,9,10
	CLO3: Recognize the importance of diversity and inclusivity for societal harmony and peaceful co-existence.	C2 (Understand)	1,6,7,9,10
Text Book(s)	1. "Civics Today: Citizenship, Economics, & You" by McGraw-Hill Education 2. "Citizenship in Diverse Societies" by Will Kymlicka and Wayne Norman. 3. "Digital Citizenship in Action: Empowering Students to Engage in Online Communities" by Kristen Mattson. 4. "Globalization and Citizenship: In the Pursuit of a Cosmopolitan Education" by Graham Pike and David Selby.		
Reference Material	1. "Community Engagement: Principles, Strategies, and Practices" by Becky J. Feldpausch and Susan M. Omilian. 2. "Creating Social Change: A Blueprint for a Better World" by Matthew Clarke and Marie- Monique Steckel.		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to Civics and Citizenship: Meaning and scope of civics, Concept of citizenship (legal vs participatory), Civic engagement: forms and importance Role of citizens in society	Book1: Chapter 1
Week 2	Historical Evolution of Civic Participation: Early forms of civic participation (tribal, communal systems), Evolution through democratic movements, Civic participation in colonial and post-colonial contexts, Civic development in Pakistan, Shifts from traditional to digital participation	Book1: Chapter 2 Case Study 1
Week 3	Types of Citizenship: Active citizenship, Participatory citizenship, Digital citizenship, Global citizenship, Rights-based vs duty-based citizenship, Comparative analysis of types	Book1: Chapter 3 Quiz 1
Week 4	Democracy and Citizenship: Core principles of democracy, Relationship between democracy and citizen participation, Role of elections and representation, Accountability and transparency, Challenges to democratic participation	Book1: Chapter 4 Assignment 1
Week 5	Foundations of Modern Society and Citizenship: Social contract theory (basic idea), Rule of law, Equality and justice, Civic values	Book1: Chapter 5

	(tolerance, respect, responsibility), Institutional trust and legitimacy	
Week 6	Structure of Government in Pakistan: Federal system overview, Legislative, executive, and judiciary roles, Provincial and local governments, Separation of powers, Governance challenges in Pakistan	Book1: Chapter 6 Quiz 2
Week 7	Civil Society and Political Participation: Definition and components of civil society, NGOs, community-based organizations, Role of media in civic life, Voting rights and electoral participation, Barriers to political participation	Book1: Chapter 7 Assignment 2
Week 8	Fundamental Rights (Constitution of Pakistan 1973): Overview of the Constitution, Key fundamental rights (freedom, equality, religion, speech), Protection mechanisms, Limitations and responsibilities, Real-life application of rights	Book1: Chapter 8
Week 9	Civic Responsibilities and Ethics: Duties of citizens (law obedience, taxation, participation), Ethical principles in civic life, Accountability and transparency, Non-violence and peaceful dialogue, Civic virtues and integrity	Book1: Chapter 9
Week 10	Concept of Community: Definition and types of community (geographic, interest-based), Characteristics of communities, Social bonds and networks, Role of culture and identity, Community challenges	Book1: Chapter 10 Case Study 2
Week 11	Community Engagement Approaches: Meaning and importance of community engagement, Participatory approaches, Bottom-up vs top-down development, Tools for engagement (meetings, surveys, digital tools), Barriers to engagement	Book1: Chapter 11 Quiz 3
Week 12	Case Studies of Community Initiatives: Overview of successful initiatives (local/global), Key success factors (leadership, participation, sustainability), Role of partnerships, Impact assessment, Lessons learned	Book1: Chapter 12
Week 13	Advocacy and Activism: Meaning of advocacy and activism, Public discourse and opinion formation, Tools of advocacy (campaigns, petitions, media), Social movements (local/global examples), Risks and ethical considerations	Book1: Chapter 13 Assignment 3
Week 14	Digital Citizenship and Cyber Ethics: Concept of digital citizenship, Responsible use of social media, Cyber ethics (privacy, misinformation, cyberbullying), Digital identity and footprint, Legal aspects of online behavior	Book1: Chapter 14
Week 15	Digital Divide and Its Impact: Meaning and dimensions of digital divide, Access, usage, and skill gaps, Socioeconomic and geographic disparities, Impact on education, employment, and civic participation, Strategies to reduce digital divide	Book1: Chapter 15 Quiz 4
Week 16	Diversity, Inclusion, and Social Justice: Understanding diversity (ethnic, cultural, economic, gender), Inclusion and equity concepts, Minority rights and representation, Social inequalities in Pakistan, Promoting inclusive citizenship, Role of youth in social harmony	Book1: Chapter 16

Course Title	**Ideology and Constitution of Pakistan**		
Course Code	**GE-261**		
Credit Hours	2 (2,0)		
Category	General Education		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	This course is designed to provide students with a fundamental exploration of the ideology and the constitution of Pakistan. The course focuses on the underlying principles, beliefs, and aspirations that have been instrumental in shaping the creation and development of Pakistan as a sovereign state. Moreover, the course will enable students to understand the core provisions of the Constitution of the Islamic Republic of Pakistan concerning the fundamental rights and responsibilities of Pakistan citizens to enable them function in a socially responsible manner.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Demonstrate enhanced knowledge of the basis of the ideology of Pakistan with special reference to the contributions of the founding father of Pakistan.	C1 (Know)	2,8
	CLO2: Demonstrate fundamental knowledge about the Constitution of Pakistan 1973 and its evolution with special reference to state structure.	C2 (Describe)	2,8
	CLO3: Explain about the guiding principles on rights and responsibilities of Pakistan citizens as enshrined in the Constitution of Pakistan 1973.	C2 (Describe)	8,9
Text Book(s)/ Reference Books	1. "The Idea of Pakistan" by Stephen P. Cohen. 2. "Ideology of Pakistan" by Javed Iqbal. 3. "The Struggle for Pakistan" by I.H. Qureshi. 4. "Pakistan the Formative Phase" by Khalid Bin Sayeed. 5. "Pakistan: Political Roots and Development" by Safdar Mahmood. 6. "Ideology of Pakistan" by Sharif-ul-Mujahid. 7. "The Struggle for Pakistan: A Muslim Homeland and Global Politics" by Ayesha Jala. 8. "Jinnah, Pakistan and Islamic Identity: The Search for Saladin" by Akbar S. Ahmed. 9. "The Making of Pakistan: A Study in Nationalism" by K.K. Aziz. 10. "Pakistan: A New History" by Lan Talbot. 11. "Pakistan in the Twentieth Century: A Political History" by Lawrence Ziring. 12. "The Constitution of Pakistan 1973". Original. 13. "Constitutional and Political Development of Pakistan" by Hamid Khan. 14. "The Parliament of Pakistan" by Mahboob Hussain. 15. "Constitutional Development in Pakistan" by G.W. Choudhury. 16. "Constitution-Making in Pakistan: The Dynamics of Political Order" by G.W. Choudhury.		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Introduction to the Ideology of Pakistan, Definition and significance of ideology.	Hussain: Chapter 1
Week 2	Historical contest of the creation of Pakistan (with emphasis on socio-political religious and cultural dynamics of British India between 1857 till 1947).	Hussain: Chapter 2
Week 3	Contributions of founding fathers of Pakistan of Pakistan in the freedom movement including but not limited to Allama Muhammad Iqbal, Muhammad Ali Jinnah., etc.	Hussain: Chapter 4-5
Week 4	Contributions of women and students in the freedom	Hussain: Chapter 8-9

	movement for separate homeland for Muslims of British India.	Quiz 1
Week 5	Two-Nation Theory, Evolution of the Two-Nation Theory (Urdu Hindi controversy, Partition of Bengal, Simla Deputation 1906,	Hussain: Chapter 10-12 Assignment
Week 6	Allama Iqbal's Presidential Address 1930, Congress Ministries 1937 Lahore Resolution 1940).	Hussain: Chapter 13-14
Week 7	Introduction to the Constitution of Pakistan: Definition and importance of a constitution.	Hussain: Chapter 29
Week 8	Ideological factors that shaped the Constitution(s) of Pakistan (Objectives Resolution 1949).	Hussain: Chapter 17
Week 9	Constitution and State Structure: Structure of Government (executive, legislature, and judiciary).	Hussain: Chapter 30
Week 10	Distribution of powers between federal and provincial governments.	Hussain: Chapter 38 Quiz 2
Week 11	18th Amendment and its impact on federalism.	Hussain: Chapter 39
Week 12	Fundamental Rights, Principles of Policy and Responsibilities: Overview of fundamental rights guaranteed to citizens by the Constitution of Pakistan 1973 (Articles 8-28).	Hussain: Chapter 35 Hussain: Chapter 35 Assignment
Week 13	Overview of Principles of Policy (Articles 2940).	Hussain: Chapter 36
Week 14	Responsibilities of the Pakistan citizens (Article 5).	Hussain: Chapter 34 Presentation/Project
Week 15	Constitutional Amendments: Procedures for amending the Constitution.	Hussain: Chapter 39 Presentation/Project
Week 16	Notable Constitutional amendments and their implications	Hussain: Chapter 39

Course Title	**Applied Physics**		
Course Code	**GE-262**		
Credit Hours	3 (2,1)		
Category	General Education (Natural Science)		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Introduction	The course introduces students with the basic concept of Physics and electronics. Students are also taught Physics laws and other associate topics to prepare them for the advanced level courses in this area. The focus of the course on electric force and its applications and related problems, conservation of charge, charge quantization, Electric fields due to point charge and lines of force and many other useful topics.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: To understand the fundamental concepts of Physics.	C2 (Understand)	1,2
	CLO2: To understand about charges and their interactions.	C2 (Understand)	1,2
	CLO3: To develop strong concepts of numerical techniques related to vectors and electrostatics and magnetism.	C2 (Understand)	1,2
	CLO4: To develop the relation between electricity and magnetism.	C4 (Analyze)	1,2,3
Text Book(s)	1. D. Halliday, R. Resnick, Kenneth S. Krane, Physics Vol. 2, 5th Ed., John Wiley, 2001, ISBN: 978-0471401940.		
Reference Material	1. Hugh D. Young, Roger A. Freedman, A. Lewis, Sears, University Physics, 11th Ed., Benjamin-Cummings Pub. Co., 2004, ISBN: 978-0805391794. 2. D. Halliday, R. Resnick, J. Walker, Fundamentals of Physics, 6th Ed., Wiley, 2010, ISBN: 978-0470469118. 3. Fundamentals of Physics (Extended), 10th edition, Resnick and Walker 4. Narciso Garcia, Arthur Damask, Steven Schwarz., "Physics for Computer Science Students", Springer Verlag, 1998.		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Electromagnetism: Historical Background, Electric Charge Quantization of Charge, Conductors and Insulators, Conservation of Charge, Coulomb's Law, Vector form of Coulomb's Law	Halliday et al.: Chapter 25
Week 2	Superposition, Continuous Charge Distributions, (Line, ring, disk)	Halliday et al.: Chapter 25 Assignment 1
Week 3	The Field, The Electric Field, The Electric Field of Point Charges, Electric Field of Continuous Charge Distributions	Halliday et al.: Chapter 26 Quiz 1
Week 4	Electric Field Lines, Electric Field of an Electric Dipole Dipole in an External Electric Field, Problems Related with Electric Field	Halliday et al.: Chapter 26 Assignment 2
Week 5	The Flux of a Vector Field, The Flux of an Electric Field Gauss' Law, Applications of Gauss' Law	Halliday et al.: Chapter 27 Assignment 3
Week 6	Electric potential energy, Electric potentials, Calculating the potential from the field and related problem, Potential due to point charge, Potential Due to Collection of Point Charges	Halliday et al.: Chapter 28 Quiz 2

Week 7	Electric Potential of Continuous Charge Distributions Calculating the Field from the Potential, Equipotential Surfaces, Electric field Lines and Equipotential Surfaces Problems	Halliday et al.: Chapter 28 Assignment 4
Week 8	Electric current, Current density, Resistance, Resistivity and conductivity, Ohm's law and its applications	Halliday et al.: Chapter 29 Assignment 5
Week 9	The magnetic force on a moving charge, The magnetic force on a current, The Hall effect	Halliday et al.: Chapter 32 Assignment 6
Week 10	The Biot- Savart law Line of B, Two parallel conductors	Halliday et al.: Chapter 33
Week 11	Amperes' s Law, Applications (Wire, Solenoid, Toroid)	Halliday et al.: Chapter 33 Quiz 3
Week 12	Faraday's experiments, Faraday's Law of Induction, Lenz's law, Motional emf, Induced electric field, The basic equation of electromagnetism, Induced Magnetic field, The displacement current	Halliday et al.: Chapter 34 Hugh et al.: Chapter 30
Week 13	Reflection and Refraction of light waves, Total internal reflection	Halliday et al.: Chapter 39
Week 14	Two source interference, Double Slit interference Related problems, Interference from thin films	Halliday et al.: Chapter 41 Assignment 7
Week 15	Diffraction and the wave theory, related problems Single-Slit Diffraction, Related problems	Halliday et al.: Chapter 42 Quiz 4
Week 16	Polarization of electromagnetic waves, Polarizing sheets Related problems, Overall Review of Course for Final Exam	Halliday et al.: Chapter 44

Course Title	**Islamic Studies**		
Course Code	**GE-264**		
Credit Hours	2 (2,0)		
Category	General Education		
Prerequisite	None		
Co-Requisite	None		
Follow Up	None		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Describe enhanced knowledge of Islamic foundational beliefs, practices, historical development, fundamental sources of Shari'ah, spiritual values, and ethical principles.	C1 (Describe)	2,8
	CLO2: Apply the basic values of Quran & Hadith in daily life.	C3 (Apply)	9,10
	CLO3: Apply the basic values of Sirah of the Holy Prophet PBUH to daily life.	C3 (Apply)	9,10
	CLO4: Identify and discuss contemporary issues being faced by the Muslim world, including social challenges, gender roles, and interfaith interactions.	C2 (Discuss)	7,8
Course Description	This course is designed to provide students with a comprehensive overview of the fundamentals aspects of Islam, its benefits, practices, history and influence on society. It will further familiarize the students with a solid foundation in understanding Islam from an academic and cultural perspective. Through this course, students will have an enhanced understanding of Islam's multifaceted dimensions which will enable them to navigate complex discussions about Islam's historical and contemporary role, fostering empathy, respect, and informed dialogue.		
Text Book(s)	Islamic Studies Compulsory book.		
Reference Material			

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	Revelation, & recitation of Holy Quran. Kinds of revelation.	Chapter 1
Week 2	Preservation of the holy Quran	Chapter 1
Week 3	Introduction of Hadith and its kinds.	Chapter 1-2 Assignment 1
Week 4	Importance & benefits of Hadith	Chapter 2
Week 5	Study of Hadith # 1,2,3,4	Chapter 2
Week 6	Study of Hadith # 5,6,7,8,9,10,11	Chapter 2-3
Week 7	Study of Hadith # 12,13,14,15,16,17	Chapter 3 Assignment 2
Week 8	Mid term exam	
Week 9	Study of Sirah of the Holy Prophet PBUH	Chapter 4

Week 10	Study of Sirah of the Holy Prophet PBUH	Chapter 4 Assignment 3
Week 11	Caliphate of Hadhrat Abu Bakr & Hadhrat Umar RA	Chapter 5
Week 12	Caliphate of Hadhrat Usman & Hadhrat Ali RA	Chapter 5
Week 13	Introduction of Islamic Fiqh	Chapter 6 Assignment 4
Week 14	Social techings of Islam, Islamic Family system	Chapter 7
Week 15	Ethical Values of Islam Islam & Globalization	Chapter 7
Week 16	Islam & Modern world, islamophobia	Chapter 8

* Please note that the above outline is the English version of the one attached in Annex A1.

Course Title	**Entrepreneurship**		
Code	**GE-265**		
Credit Hours	2 (2,0)		
Category	General Education		
Prerequisite	None		
Co-Requisite	None		
Follow-up	None		
Course Introduction	This course is designed to promote entrepreneurial spirit and outlook among students, encouraging them to think critically, identify opportunities, and transform their ideas into successful ventures. It aims at imparting them with the requisite knowledge; skills and abilities, enabling them seize the identified opportunities for initiating of business (including requirements for registration and incorporation with regulators such as SECP and others), market research, opportunity identification, business planning, financial literacy for managing finances and securing funding, marketing and sales, team building and innovation, overall, the course is geared towards personal growth and professional development for pursuing innovative ideas, availing opportunities and initiating start-ups.		
Course Learning Outcomes (CLOs)	At the end of the course, the students will be able to:	**BT**	**PLO**
	CLO1: Understand key entrepreneurial concepts, including the different stages of the entrepreneurial process as well as specific regulatory requirements to set up an enterprise in Pakistan.	C2 (Understand)	1,8
	CLO2: Apply entrepreneurial frameworks and techniques to identify business opportunities and use this knowledge to produce practical solutions to real-world startup challenges.	C3 (Apply)	2,3,4
	CLO3: Develop a comprehensive business plan for a new venture by analyzing the feasibility of a new business idea and by justifying the proposed strategies.	C6 (Create)	3,4,7,10
Practical Requirements	As part of the overall learning requirements, students shall be tasked with creating and presenting a comprehensive business plan at the end of the course for a hypothetical or real business idea. This practical exercise shall allow them to apply the knowledge, skills and abilities acquired in the course to develop a feasible business plan and where possible explore the possibility of implementing the plan with support and assistance from established business-persons and entrepreneurs.		
Textbook	1. B. R. Barringer, and R. D. Ireland, Entrepreneurship: Successfully Launching New Ventures, 6th Edition, Prentice Hall, 2019, ISBN: 978-0134729534.		
Suggested Instructional/ Reading Material	1. "Entrepreneurship: Theory, Process, and Practice" by Donald F. Kuratko. 2. "New Venture Creation: Entrepreneurship for the 21st Century" by Jeffry A. Timmons, Stephen Spinelli Jr., and Rob Adams. 3. "Entrepreneurship: A Real-World Approach" by Rhonda Abrans. 4. "The Lean Startup: How Today's Entrepreneurs use Continuous Innovation to Create Radically Successful Businesses" by Eric Ries. 5. "Effectual Entrepreneurship" by Stuart Read, Saras Sarasvathy, Nick Dew, Robert Wiltbank, and Anne-Valcric Ohlsson.		

Weekly Lesson Plan

Week No.	Course Content	Assignments/Readings
Week 1	What is Entrepreneurship, and Why is it important? Why do People Become Entrepreneurs? Characteristics of Successful Entrepreneurs, Types of Start-up Firms	Barringer et al.: Chapter 1
Week 2	The Positive Effects of Entrepreneurship and Entrepreneurial Firms, The Entrepreneurial Process	Barringer et al.: Chapter 1-2

	The differences between opportunities and ideas	Case Study 1
Week 3	Three Ways to Identify Opportunities, Personal Characteristics of the Entrepreneur, Techniques for generating ideas, Encouraging the development of new ideas	Barringer et al.: Chapter 2 Quiz 1
Week 4	Feasibility Analysis, Product/Service Feasibility Analysis, Industry/Target Market Feasibility Analysis, Organizational Feasibility Analysis, Financial Feasibility Analysis	Barringer et al.: Chapter 3 Assignment 1 (Concept Statement)
Week 5	Business Models and their Importance, General Categories of Business Models, The Barringer/Ireland Business Model Template	Barringer et al.: Chapter 4 Assignment 2 (Feasibility Analysis)
Week 6	Industry Analysis, The Five Forces Model, The Value of the Five Forces Model, Industry Types and The Opportunities They Offer, Competitor analysis	Barringer et al.: Chapter 5 Quiz 2
Week 7	The Business Plan: Who reads the Business Plan? What are they looking for? Guidelines for Writing a Business Plan, Outline of the Business Plan, Presenting the Business Plan to Investors	Barringer et al.: Chapter 6 Assignment 3 (Business Model)
Week 8	Revision and Discussion on Students' Business Ideas	Barringer et al.: Chapter 1-6
Week 9	Introduction to Financial Management, Financial Objectives of a Firm, The Process of Financial Management	Barringer et al.: Chapter 8
Week 10	Financial Statements (Balance Sheet, Income Statement, Cash Flow Statement), Forecasts, Pro Forma Financial Statements	Barringer et al.: Chapter 10 Case Study 2
Week 11	Liability of newness as a Challenge, Creating a new-venture team, Rounding out the team: the role of Professional Advisers, Other Professionals	Barringer et al.: Chapter 9 Quiz 3
Week 12	Selecting a Market and Establishing a Position, Branding	Barringer et al.: Chapter 11 Case Study 3
Week 13	The 4Ps of marketing for new ventures, Sales Process, and Related Issues	Barringer et al.: Chapter 11 Assignment 4 (Business Plan)
Week 14	Final Project Presentation	Barringer et al.: Chapter 1-11
Week 15	Internal Growth Strategies, Additional Internal product-growth Strategies, International Expansion, External Growth Strategies	Barringer et al.: Chapter 13 Quiz 4
Week 16	Revision and Wrap-Up of the Course	Barringer et al.: Chapter 1-13

Annex-A1