



PUNYASHLOK AHILYADEVJI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR

FACULTY OF ENGINEERING & TECHNOLOGY

INFORMATION TECHNOLOGY

Syllabus Structure for

F.Y. B.Tech. (All Branches) w.e.f. Academic Year 2018-19

S.Y. B.Tech. (Information Technology) w.e.f. Academic Year 2019-20

T.Y. B.Tech. (Information Technology) w.e.f. Academic Year 2020-21

Final Year B.Tech. (Information Technology) w.e.f. Academic Year 2021-22

Choice Based Credit System

॥ विद्या संपन्नता ॥



PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
FACULTY OF ENGINEERING & TECHNOLOGY
Information Technology

Program Educational Objectives and Outcomes

A. Program Educational Objectives

1. Graduates will exhibit strong fundamental knowledge and skills in the field of Information Technology to pursue successful professional careers and higher studies and research.
2. Graduates will exhibit capabilities to understand and resolve the various societal issues through their problem solving skills.
3. Graduates will be sensitive to ethical, societal and environmental issues while serving at their professional work and society.

B. Program Outcomes

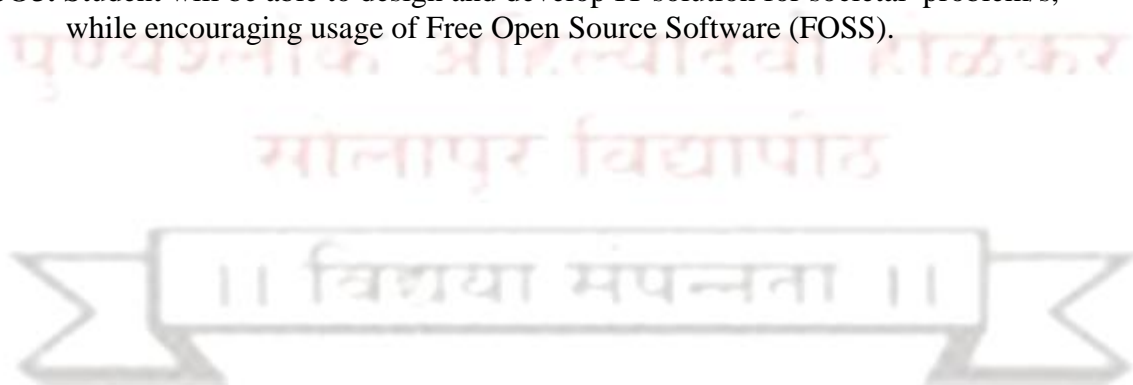
Engineering Graduate will be able to –

4. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
5. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
6. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
7. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
8. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
9. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
10. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

- 11. Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
- 12. Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
- 13. Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
- 14. Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
- 12. Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes (PSOs)

- PSO1.** Student will be able to apply fundamentals of mathematics, algorithms and computational systems to Information Technology.
- PSO2.** Student will be able to provide a solution to the problem in the areas of Networking, Database management, System Software, Web Technology, Information Security and Thrust areas..
- PSO3.** Student will be able to design and develop IT solution for societal problem/s, while encouraging usage of Free Open Source Software (FOSS).





PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
FACULTY OF ENGINEERING & TECHNOLOGY

CBCS Curriculum for First Year B. Tech. (All Branches)

WEF 2018-19

□ Semester I : Theory Courses

Course Code	Name of the Course	Engagement Hours			Credits	FA		SA		Total
		L	T	P		ESE	ISE	ICA		
C011/ C012	Engineering Physics / Engineering Chemistry\$	3			3	70	30			100
C112	Engineering Mathematics I	3			3	70	30			100
C113	Basic Electrical & Electronics Engineering	4			4	70	30			100
C114	Engineering Mechanics	3			3	70	30			100
C115	Basic Mechanical Engineering	3			3	70	30			100
C116	Communication Skills	1			1		25			25
Total		17			17	350	175			525

□ Semester I : Laboratory / Tutorial Courses

Course Code	Name of the Course	Engagement Hours			Credits	FA		SA		Total
		L	T	P		ESE	ISE	ICA		
C011/ C012	Engineering Physics / Engineering Chemistry\$			2	1			25		25
C112	Engineering Mathematics I		1		1			25		25
C113	Basic Electrical & Electronics Engineering			2	1			25		25
C114	Engineering Mechanics			2	1			25		25
C115	Basic Mechanical Engineering			2	1			25		25
C116	Communication Skills			2	1			25		25
C117	Workshop Practice			2	1			25		25
Total			1	12	7			175		175
Grand Total		17	1	12	24	350	175	175		700
C118	Induction Program	# (Please see note below)								

□ Semester II : Theory Courses

Course Code	Name of the Course	Engagement Hours			Credits	FA	SA		Total
		L	T	P		ESE	ISE	ICA	
C011/ C012	Engineering Physics / Engineering Chemistry\$	3			3	70	30		100
C122	Engineering Mathematics II	3			3	70	30		100
C123	Engineering Graphics & Design	3			3	70	30		100
C124	Basic Civil Engineering	3			3	70	30		100
C125	Programming for Problem Solving	2			2		25		25
C126	Professional Communication	1			1		25		25
Total		15			15	280	170		450
C127	Democracy, Elections and Good Governance					30			30

□ Semester II : Laboratory / Tutorial Courses

Course Code	Name of the Course	Engagement Hours			Credits	FA	SA		Total
		L	T	P		ESE (POE)	ISE	ICA	
C011/ C012	Engineering Physics / Engineering Chemistry\$			2	1			25	25
C122	Engineering Mathematics II		1		1			25	25
C123	Engineering Graphics & Design			4	2			50	50
C124	Basic Civil Engineering			2	1			25	25
C125	Programming for Problem Solving			4	2	50#		50	100
C127	Professional Communication			2	1			25	25
Total			1	14	8	50		200	250
Grand Total		15	1	14	23	330	170	200	700
C128	Democracy, Elections and Good Governance							20	

□ Legends used –

L	Lecture	FA	Formative Assessment
T	Tutorial	SA	Summative Assessment
P	Lab Session	ESE	End Semester Examination
		ISE	In Semester Evaluation
		ICA	Internal Continuous Assessment

□ Notes-

1. \$ - Indicates approximately half of the total students at F.Y B..Tech will enroll under Group A and remaining will enroll under Group B.

Group A will take up course of Engineering Physics (theory & laboratory) in Semester I and will take up course of Engineering Chemistry (theory & laboratory) in semester II.

Group B will take up course of Engineering Chemistry (theory & laboratory) in Semester I and will take up course of Engineering Physics (theory & laboratory) in semester II

2. # - Indicates the subject 'Programming for Problem Solving' shall have a University 'Practical and Oral Examination' at the end of the semester assessing student's programming skills.

3. In Semester Evaluation (ISE) marks shall be based upon student's performance in minimum two tests & mid-term written test conducted & evaluated at institute level

Internal Continuous Assessment Marks (ICA) are calculated based upon student's performance during laboratory sessions / tutorial sessions

4. Democracy, Elections & Good Governance is mandatory course. The marks earned by student with this course shall not be considered for calculation of SGPA/CGPA. However student must complete ICA of 20 marks and End Semester Examination (ESE) of 30 marks (as prescribed by university, time to time) for fulfillment of this course. This course is not considered as a passing head for counting passing heads for ATKT. However, student must pass this subject for award of the degree

5. Student must complete induction program of minimum five days before commencement of the regular academic schedule at the first semester.

GUIDELINES FOR INDUCTION PROGRAM (C128)

New entrants into an Engineering program come with diverse thoughts, mind set and different social, economical, regional and cultural backgrounds. It is important to help them adjust to the new environment and inculcate in them the ethos of the institution with a sense of larger purpose.

A **Five day** induction program for the new UG entrant students is proposed at the commencement of the first semester. It is expected to complete this induction program before commencement of the regular academic schedule.

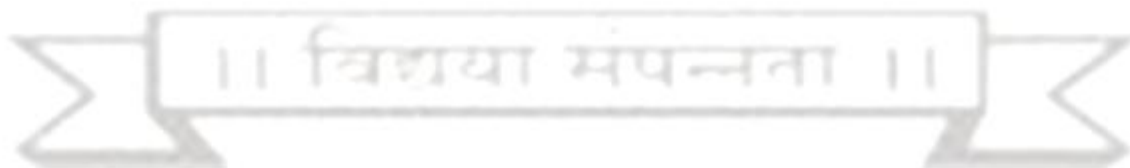
Its purpose is to make new entrants comfortable in their new environment, open them up, set a healthy daily routine for them, create bonding amongst the peers as well as between faculty and students, develop awareness, sensitivity and understanding of the self, people around them, society at large, and nature.

The Induction Program shall encompass (but not limited to) below activity –

1. Physical Activities
2. Creative Arts
3. Exposure to Universal Human Values
4. Literary Activities
5. Proficiency Modules
6. Lectures by Experts / Eminent Persons
7. Visit to Local Establishments like Hospital / Orphanage
8. Familiarization to Department

Induction Program Course do not have any marks or credits however performance of students for Induction Program is assessed at institute level using below mandatory criteria –

1. Attendance and active participation
2. Report writing





PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Engineering & Technology (Revised from 2018-19)

Credit System structure of S.Y. B. Tech. Information Technology W.E.F. 2019-20
Semester I

Course Code	Theory Course Name	Hrs./week			Credits	Examination Scheme				
		L	T	P		ISE	ESE	ICA	Total	
IT211	Applied Mathematics - I	3	1	--	4	30	70	25	125	
IT212	Discrete Mathematical Structure	3	1	--	4	30	70	25	125	
IT213	Data Communication	3	--	--	3	30	70	--	100	
IT214	Digital Techniques	4	--	--	4	30	70	--	100	
IT215	Computer Graphics	3	--	--	3	30	70	--	100	
IT216	Advanced C Concepts	2	-	-	2	25	-	-	25	
	Sub Total	18	2	--	20	175	350	50	575	
EN V21	Environmental Science - I	1	-	-	-	-	-	-	-	
Course Code	Laboratory Course Name									
							<i>ESE</i>			
							<i>POE</i>	<i>OE</i>		
IT213	Data Communication	--	--	2	1	--	50	--	25	75
IT214	Digital Techniques	--	--	2	1	--	50	--	25	75
IT215	Computer Graphics	--	--	2	1	--		--	25	25
IT216	Advanced C Concepts	--	--	4	2	--	50	--	25	75
	Sub Total	--	--	10	5	--	150		100	250
	Grand Total	18	2	10	25	175	500		150	825

- Abbreviations: L- Lectures, P –Practical, T- Tutorial, ISE- In Semester Exam, ESE - End Semester Exam , ICA- Internal Continuous Assessment, ESE - University Examination (Theory &/ POE &/Oral examination)

Note

1. Student is required to study and pass Environmental Science subject in Second Year of Engineering to become eligible for award of degree.
2. Batch size for the practical /tutorial shall be of 20 students. On forming the batches, if the strength of remaining students exceeds 9, then a new batch shall be formed.
3. Vocational Training (evaluated at Final Year B. Tech. Part-I) of minimum 15 days shall be

completed in any vacation after S.Y. B.Tech. Part-II but before Final Year B.Tech. Part-I & the report shall be submitted and evaluated in Final Year B.Tech. Part-I

4. Student shall select one Self Learning Module at T.Y. B.Tech Part I and T.Y. B.Tech Part II each from Technical and Humanities and Social Sciences Group with at least one Self Learning Module from the Humanities and Social Sciences Group
5. Curriculum for Humanities and Social Sciences Self Learning Modules is common for all undergraduate programs of faculty of Engineering and Technology
6. ICA assessment shall be a continuous process based on student's performance in – class tests, assignments, homework, subject seminars, quizzes, laboratory books and their interaction and attendance for theory and lab sessions as applicable





PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Engineering & Technology (Revised from 2018-19)

Credit System structure of S.Y. B.Tech. Information Technology W.E.F. 2019-20

Semester II

Course Code	Theory Course Name	Hrs./week			Credits	Examination Scheme				
		L	T	P		ISE	ESE	ICA	Total	
IT221	Applied Mathematics – II	3	1	--	4	30	70	25	125	
IT222	Theory of Computation	4	1	--	5	30	70	25	125	
IT223	Microprocessor	3	--	--	3	30	70	--	100	
IT224	Data Structures	3	--	--	3	30	70	--	100	
IT225	Computer Networks	3	--	--	3	30	70	--	100	
IT226	Object Oriented Programming Through C ++	2	--	--	2	25	--	--	25	
	Sub Total	18	2	--	20	175	350	50	575	
ENV22	Environmental Science - II	1	-	-	-	-	-	-	1	
Course Code	Laboratory Course Name									
							ESE			
							POE	OE		
IT223	Microprocessor	--	--	2	1	--	50	--	25	75
IT224	Data Structures	--	--	4	2	--	50	--	25	75
IT225	Computer Networks	--	--	2	1	--	-	--	25	25
IT226	Object Oriented Programming Through C ++	--	--	2	1	--	50	--	25	75
	Sub Total			10	5		150		100	250
	Grand Total	18	2	10	25	175	500	100	825	

- Abbreviations: L- Lectures, P –Practical, T- Tutorial, ISE- In Semester Exam, ESE - End Semester Exam, ICA- Internal Continuous Assessment, ESE - University Examination (Theory &/ POE &/Oral examination)

Note

1. Student is required to study and pass Environmental Science subject in Second Year of Engineering to become eligible for award of degree.
2. Batch size for the practical /tutorial shall be of 20 students. On forming the batches, if the strength of remaining students exceeds 9, then a new batch shall be formed.
3. Vocational Training (evaluated at Final Year B. Tech. Part-I) of minimum 15 days shall be completed in any vacation after S.Y. B.Tech. Part-II but before Final Year B.Tech. Part-I & the report shall be submitted and evaluated in Final Year B.Tech. Part-I

4. Student shall select one Self Learning Module at T.Y. B.Tech Part I and T.Y. B.Tech Part II each from Technical and Humanities and Social Sciences Group with at least one Self Learning Module from the Humanities and Social Sciences Group
5. Curriculum for Humanities and Social Sciences Self Learning Modules is common for all undergraduate programs of faculty of Engineering and Technology
6. ICA assessment shall be a continuous process based on student's performance in – class tests, assignments, homework, subject seminars, quizzes, laboratory books and their interaction and attendance for theory and lab sessions as applicable



पुण्यश्लोक अहिल्यादेवी संसृष्टकर
सांलापुर विद्यापीठ

॥ विशया मंपन्नता ॥



PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Engineering & Technology (Revised from 2018-19)

Credit System structure of T.Y. B. Tech. Information Technology W.E.F. 2020-21

Semester I

Course Code	Theory Course Name	Hrs./week			Credits	Examination Scheme				
		L	T	P		ISE	ESE	ICA	Total	
IT311	Operating System	3	--	--	3	30	70	--	100	
IT312	System Software	4	--	--	4	30	70	--	100	
IT313	Design and Analysis of Algorithms	3	1	--	3	30	70	25	125	
IT314	Database Engineering	3	--	--	3	30	70	--	100	
IT315	Computer Organization and Architecture	3	1	--	3	30	70	25	125	
IT316	Java Programming	2	-	-	2	25	-	-	25	
SLH31	Self Learning Module-I	--	--	--	2	--	50	--	50	
	Sub Total	18	2	-	20	175	400	50	625	
Course Code	Laboratory Course Name									
							ESE			
							POE	OE		
IT311	Operating System	--	--	2	1	--	50	--	25	75
IT312	System Software	--	--	2	1	--	--	--	25	25
IT314	Database Engineering	--	--	2	1	--	50	--	25	75
IT316	Java Programming	--	--	4	2	--	50	--	25	75
	Sub Total	--	-	10	5	--	150	75	250	
	Grand Total	18	2	10	25	175	550	150	875	

- Abbreviations: L- Lectures, P –Practical, T- Tutorial, ISE- In Semester Exam, ESE - End Semester Exam, ICA- Internal Continuous Assessment, ESE - University Examination (Theory & POE &/Oral examination)



PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Engineering & Technology (Revised from 2018-19)
Credit System structure of T.Y. B. Tech. Information Technology W.E.F. 2020-21

Semester II

Course Code	Theory Course Name	Hrs./week			Credits	Examination Scheme				
		L	T	P		ISE	ESE	ICA	Total	
IT321	Unix Operating System	4	--	--	4	30	70	--	100	
IT322	Software Engineering	3	1	--	4	30	70	25	125	
IT323	Object Oriented Modeling and Design	3	1	--	4	30	70	25	125	
IT324	Artificial Intelligence	3	--	--	3	30	70	--	100	
IT325	Mobile Application Development	3	--	--	3	30	70	--	100	
IT326	Programming Elective - I	2	-	-	2	25	-	-	25	
IT327	Self Learning Module-II	--	--	--	2	--	50	--	50	
	Sub Total	18	2	--	22	175	400	50	625	
Course Code	Laboratory Course Name									
							ESE			
							POE	OE		
IT321	Unix Operating System	--	--	2	1	--	--	25	25	50
IT324	Artificial Intelligence	--	--	2	1	--	--	--	25	25
IT325	Mobile Application Development	--	--	2	1	--	50	_	25	75
IT326	Programming Elective - I	--	--	2	1	--	50	_	25	75
IT328	Seminar	--	--	2	1	--	--	--	25	25
	Sub Total		--	10	5	--	125		125	250
	Grand Total	18	2	10	27	175	525		175	875

- Abbreviations: L- Lectures, P –Practical, T- Tutorial, ISE- In Semester Exam, ESE - End Semester Exam, ICA- Internal Continuous Assessment, ESE - University Examination (Theory &/ POE &/Oral examination)

Programming Elective – I

IT315A : Python

IT316B : Angular JS

1. Batch size for the practical /tutorial shall be of 15 students. On forming the batches, if the strength of remaining student exceeds 7, then a new batch shall be formed.
2. Vocational Training (evaluated at Final Year B.Tech Part-I) of minimum 15 days shall be completed in any vacation after S.Y B.Tech. Part-II but before Final Year B.Tech. Part-I & the report shall be submitted and evaluated in Final Year B.Tech Part-I
3. Student shall select one Self Learning Module at T.Y. B.Tech Part I and T.Y .B.Tech. Part II each from Technical and Humanities and Social Sciences Group with at least one Self Learning Module from the Humanities and Social Sciences Group
4. Curriculum for Humanities and Social Sciences Self Learning Modules is common for all under graduate programs of faculty of Engineering and Technology
5. Minimum four assignments for Self Learning Modules at T.Y. B.Tech. Part I and T.Y .B.Tech Part II shall be submitted by the students which shall be evaluated by a Module Coordinator assigned by institute / department
6. Project groups for Final Year B.Tech Part I and Part II formed at T.Y. B.Tech Part II for seminar shall not be of more than **five** students.
7. ICA assessment shall be a continuous process based on student's performance in – class tests, assignments, homework, subject seminars, quizzes, laboratory books and their interaction and attendance for theory and lab sessions as applicable

पुण्यश्लोक अहिल्यादेवी संककर
सांलापुर विद्यापीठ

॥ विशया मंपन्नता ॥



PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Engineering & Technology (Revised from 2018-19)

Credit System structure of Final Year B.Tech. Information Technology W.E.F. 2021-22

Semester I

Course Code	Theory Course Name	Hrs./week			Credits	Examination Scheme				
		L	T	P		ISE	ESE	ICA	Total	
IT411	Machine Learning	3	--	--	3	30	70	--	100	
IT412	Advanced Database System	4	--	--	4	30	70	--	100	
IT413	Software Testing and Quality Assurance	3	-	--	3	30	70	25	125	
IT414	Distributed Computing	3	--	--	3	30	70	--	100	
IT415A to IT415B	Elective-I	3	--	--	3	30	70	25	125	
IT416A To IT416B	Programming Elective - II	2	--	--	2	25	--	--	25	
IT417	Project-I	--	4	--	4	--	--	75	75	
	Sub Total	18	4	--	22	175	350	125	650	
Course Code	Laboratory Course Name									
							ESE			
							POE	OE		
IT411	Machine Learning	--	--	2	2	--	--	--	25	25
IT412	Advanced Database System	--	--	2	2	--	50	--	25	75
IT414	Distributed Computing	--	--	2	2	--	--	--	25	25
IT416	Programming Elective - II	--	--	2	2	--	50	--	25	75
IT418	Vocational Training	--	--	--	--	--	--	--	25	25
	Sub Total	--	--	8	8	--	100	125	225	
	Grand Total	18	4	8	30	175	450	250	850	

- Abbreviations: L- Lectures, P –Practical, T- Tutorial, ISE- In Semester Exam, ESE - End Semester Exam, ICA- Internal Continuous Assessment, ESE – University Examination (Theory &/ POE &/Oral examination)

Programming Elective – II

IT416A : C# .Net

IT416B : Web Technology



PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR

Faculty of Engineering & Technology (Revised from 2018-19)

Credit System structure of Final Year B.Tech. Information Technology W.E.F. 2021-22

Semester II

Course Code	Theory Course Name	Hrs./week			Credits	Examination Scheme				
		L	T	P		ISE	ESE	ICA	Total	
IT421	Management Information System	4	--	--	4	30	70	--	100	
IT422	Information Assurance and Security	4	--	--	4	30	70	25	125	
IT423A to IT 423B	Elective –II	3	2	--	5	30	70	25	125	
IT424A to IT424B	Elective –III	3	2	--	5	30	70	25	125	
IT425A To IT425B	Programming Elective - III	2	--	--	2	25	--	--	25	
	Sub Total	16	4	--	20	145	280	75	500	
Course Code	Laboratory Course Name									
							ESE			
							POE	OE		
IT421	Management Information System	--	--	2	1	--	50	--	25	75
IT423A To IT423B	Elective - II	--	--	2	1	--	--	--	25	25
IT425A To IT425B	Programming Elective - III	--	--	2	1	--	50	--	25	75
IT426	Project-II	--	--	6	3	--	100	--	75	175
	Sub Total	--	--	12	6	--	200	150	350	
	Grand Total	16	4	12	26	145	480	225	850	

- Abbreviations: L- Lectures, P –Practical, T- Tutorial, ISE- In Semester Exam, ESE - End Semester Exam, ICA- Internal Continuous Assessment, ESE - University Examination (Theory &/ POE &/Oral examination)

Elective I		Elective II		Elective - III	
Course Code	Course	Course Code	Course	Course Code	Course
IT415A	Natural Language Processing	IT423A	Data Mining	IT424A	Image Processing
IT415B	Cloud Computing	IT423B	Information Retrieval	IT424B	Internet of Things

Programming Elective – IV

IT425A : R – Programming

IT425B : DevOps

Note

1. Batch size for the practical /tutorial shall be of 15 students. On forming the batches, if the strength of remaining students exceeds 7, then a new batch shall be formed.
2. Vocational Training (evaluated at Final Year B.Tech Part-I) of minimum 15 days shall be completed in any vacation after S.Y. B.Tech Part-II but before Final Year B.Tech Part-I & the report shall be submitted and evaluated in Final Year B.Tech. Part-I
3. Appropriate Elective I & II Subjects may be added when required.
4. Project group for Final Year B.Tech. (Information Technology) Part I and Part II shall not be of more than **five** students.
5. ICA assessment shall be a continuous process based on student's performance in – class tests, assignments, homework, subject seminars, quizzes, laboratory books and their interaction and attendance for theory and lab sessions as applicable

पुण्यश्लोक अहिल्यादेवी संककर
सांलापुर विद्यापीठ

॥ विशया मंपन्नता ॥



**PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
S.Y.B.TECH (INFORMATION TECHNOLOGY)**

Semester-III

IT 211 APPLIED MATHEMATICS-I

Teaching Scheme:

Lectures- 3 Hours / week, 3 Credits

Tutorial - 1 Hours / week, 1 Credit

Examination Scheme

ESE- 70 Marks

ISE - 30 Marks

ICA- 25 Marks

Introduction:

This course includes mathematical theory and concepts required by the computer engineer.

The course consists of linear differential equations which can be used for mathematical model which are appearing in computer engineering, where these variables are dynamically related. This course introduces Z- transform which provide a mathematical framework for a series of mathematical conversions that are useful for digital filters. Laplace transforms is another powerful mathematical tool for engineering problems in Computer Science and Engineering. This course also introduces Fourier series, which plays an important role in designing, and analyzing communication system. This course also introduces fundamentals of probability distributions which are useful for digital communication. This course introduces Queuing systems which are prevalent throughout society. There are applications of queuing theory in several disciplines.

Course Prerequisite:

Fundamentals of trigonometry, method of finding roots of algebraic equations, differentiation, integration, partial fraction, sum of sequence and methods of solving definite integrations, basics of statistics and probability theory

Course Objectives:

1. To introduce to student method of solving higher order linear differential equations
2. To introduce to student Laplace and inverse Laplace transforms
3. To introduce to student Fourier series of periodic functions
4. To make student understand Z transform and its properties
5. To introduce to student various probability distributions
6. To introduce to student Queuing theory and its model.

Course Outcomes:

At the end of course, students will be able to

1. Solve higher order linear differential equation with constant coefficient.
2. Apply Laplace and inverse Laplace transforms for solving linear differential equations.
3. Express a function in terms of sine's and cosines components so as to model simple periodic functions.
4. Solve problems on Z transform and explain its properties.
5. Find the relation between two variables for the given data using regression and explain various probability distribution functions.
6. Solve problems based on queuing theory.

SECTION – I

Unit 1: Linear differential equations with constant coefficients:

No. of lectures-07

Basic definition, differential operator, complimentary functions, particular integral, Shortcut methods for standard functions like e^{ax} , $\sin(ax + b)$, $\cos(ax + b)$, x^m , $e^{ax}V$ and xV , particular integral by general method (without method of variation of parameters) for other functions.

Unit 2: Z-Transform:

No. of lectures-05

Introduction, Z-Transform of standard sequence, properties of Z-transform – linearity, change of scale, shifting property, multiplication by k, division by k, inverse Z- transform –power series method, partial fraction method

Unit 3: Laplace transform:

No. of lectures-09

Definition, Laplace transform of standard functions, properties- first shifting, change of scale, multiplication of power t and division by t, Laplace transform of derivative and integral, Laplace transform of periodic functions, unit step functions and unit impulse functions, properties of inverse Laplace transforms- linear property, first shifting theorem, partial fraction, inverse transform of logarithmic & inverse trigonometric functions and convolution theorem, solution of differential equations by Laplace transform.

SECTION-II

Unit 4: Fourier series:

No. of lectures-07

Introduction, Definition, Euler's formula, Fourier series of periodic functions with period 2π and $2L$, Dirichlet's theorem (only statement), even and odd functions, half range sine and cosine series.

Unit 5: Statistics and probability:

No. of lectures-08

Coefficient of correlation by Karl Pearson's method and lines of regression of bivariate data, random variable, discrete and continuous random variable, probability density function, Binomial, Poisson, Normal distribution

Unit 6: Queuing Theory:

No. of lectures-06

Introduction, Queuing system, Distributions in Queuing systems, Kendall's notation, classification of queuing models, M/M/1: ∞/∞ models, M/M/1: N/ ∞ models.

Internal Continuous Assessment (ICA):

ICA shall consist of minimum six to eight assignments based on entire curriculum

Text books:

1. A textbook of Applied Mathematics Vol. II and Vol. III, J.N. and P.N. Wartikar, Vidyarthi Grah Prakashan, Pune.
2. Higher Engineering Mathematics, Dr.B.S. Grewal, Khanna Publications, Delhi.
3. A Textbook of Applied Mathematics, N.P. Bali, Ashok Saxena and N.Ch. S.N. Iyengar, Laxmi Publications, Delhi.
4. Advanced Engineering Mathematics, Kreyzig-John Wiley & SMS, New York.

Reference Books:

1. Advanced Engineering Mathematics, Peter O'Neil, Cengage Learning.
2. Engineering Mathematics, Srimanta Pal, Subodh Chandra Bhunia, Oxford University Press



**PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
S.Y. B.TECH (INFORMATION TECHNOLOGY)**

Semester-III

IT 212 - DISCRETE MATHEMATICAL STRUCTURES

Teaching Scheme

Lectures– 3Hours/week, 3 Credits

Tutorial – 1 Hour/week, 1 Credits

Examination Scheme

ESE – 70Marks

ISE – 30Marks

ICA– 25 Marks

Introduction:

This course introduces discrete mathematics which deals with fundamentals of mathematical reasoning and set theory. The course also introduces theoretical and mathematical aspects of relations, functions, algebraic system & Boolean algebra.

Course Prerequisite: Student shall have knowledge of basic mathematics.

Course Objectives:

1. To study fundamentals of Discrete Mathematical Structures including Mathematical Logic.
2. To acquire in-depth knowledge of Inference Theory and different methods to arrive at an inference from the given premises.
3. To study principles of Sets, Relations and Functions.
4. To get acquainted with different Algebraic Systems and develop an ability to select appropriate for given application.

Course Outcomes:

At the end of course, students will be able to

1. Arrive at an inference from the given premises applying mathematical logic.
2. Demonstrate principles of set theory, relations and functions.
3. Classify algebraic systems based on its properties and select appropriate for given application.

SECTION-I

UNIT-1	Mathematical logic	6 Hrs.
	Introduction, statements and Notation, Connectives - negation, conjunction, disjunction, conditional, bi conditional, statement formulas and truth tables, well-formed formulas, Tautologies, Equivalence of formulas, Duality law, Tautological implications, functionally complete sets of connectives, other connectives.	
UNIT-2	Representation of expressions	4 Hrs.
	Normal & Principle normal forms, completely parenthesized infix & polish notations, Theory of inference for statement calculus.	
UNIT-3	Set theory	4 Hrs.
	Basic concepts of set theory, types of operations on sets, ordered pairs, Cartesian product.	
UNIT - 4	Relations	7 Hrs.
	Relations, Properties of binary relations, Matrix and graph representation, Partition and covering of set, Equivalence relation, Composition, POSET and Hasse diagram.	

SECTION II

UNIT- 5	Functions	4 Hrs.
	Function -types, Composition of functions, Inverse functions.	
UNIT- 6	Algebraic systems	7 Hrs.
	Algebraic systems, semi groups and monoids, properties and example.	
UNIT- 7	Groups	6 Hrs.
	Polish expressions and their compilation, Groups, group codes.	
UNIT - 8	Lattices and Boolean algebra	7 Hrs.
	Lattice as POSETs, definition, examples and Properties, Special Lattices, Boolean algebra definition and examples, Boolean functions.	

Internal Continuous Assessment (ICA):

In tutorial session, students of different batches should be assigned exercise problems and should be guided for the solution. Minimum one tutorial per unit is expected.

Text Books:

1. Discrete mathematical structures with applications to computer science -- J. P. Tremblay & R. Manohar (MGH International)

Reference Books:

1. Discrete Mathematics with combinatorics and graph theory- S. SNTHA (CENGAGE Learning)
2. Discrete Mathematical Structures – Bernard Kolman, Robert C. Busby (Pearson Education)
3. Discrete mathematics -- Liu (MGH)
4. Theory and problems in Abstract algebra -- Schaums outline series (MGH)
5. Discrete Mathematical Structures- Y N Singh (WILEY)
6. Discrete Mathematics and Its Applications, Chakraborty & Sarkar, Oxford
7. Discrete Structures, S.B. Singh, Khanna Book Publishing, Delhi
8. Discrete Mathematics, T. Veerarajan, Tata McGraw-Hill





PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
S.Y.B.TECH (INFORMATION TECHNOLOGY)

Semester-III

IT 213 – DATA COMMUNICATION

Teaching Scheme

Lectures– 3 Hours/week, 3 Credits

Practical – 2 Hour/week, 1 Credits

Examination Scheme

ESE – 70Marks

ISE – 30Marks

ICA – 25 Marks

POE– 50 Marks

Introduction:

This course introduces Data Communication Fundamentals and Computer Networks basic knowledge. It also covers the details of Physical Layer, Data Link Layer and Network Layer design issues.

Course Prerequisite: Students should have knowledge of Computer Systems and basics of C programming language.

Course Objectives:

1. To introduce Fundamentals of Data Communication.
2. To study the functions of data link layer for Data transfer
3. To gain in-depth knowledge of medium access, Ethernet standards and communication devices.
4. To study the working of network layer routing algorithm

Course Outcomes:

At the end of course, students will be able to

1. Identify communication modes & network reference models for communication
2. Implement data transfer using Data Link Layer functions
3. Select medium access, IEEE standards and devices for communication.
4. Simulate & analyze functionalities of network layer routing algorithms

SECTION-I

UNIT-1	Data Communication Fundamentals	7 Hrs.
	Concepts & Terminology, Analog & Digital data transmission, Transmission impairments, Channel capacity, guided transmission media, Digital data to digital signal encoding, Asynchronous and Synchronous Transmission.	
UNIT-2	Reference Models	7 Hrs.
	Uses of Computer network, Network hardware, Network software, Types of Network Topologies, OSI reference model, TCP/IP protocol, ATM reference model.	
UNIT-3	Data Link Layer	8 Hrs.
	DLL design issues, Error detection & correction, Elementary DLL protocols, Sliding window protocols.	

SECTION-II

UNIT- 4	Medium Access Control	8 Hrs.
	Channel allocation problems, Multiple access protocol: ALOHA, CSMA, CSMA/CD, Collision free protocols, Limited contention protocols.	
UNIT- 5	IEEE standards	6 Hrs.
	IEEE standards 802.3, 802.4, 802.5 & 02.6(DQDB), Bridges, Switches, Routers	
UNIT- 6	Network Layer	9 Hrs.
	Network layer design issues, Routing algorithms: shortest path routing, flooding, flow-based routing, distance vector routing, link state routing, hierarchical routing, Congestion control algorithms, Internetwork	

Internal Continuous Assessment (ICA):

Student should perform minimum 8 experiments based on the following guidelines.

1. Implementation of simplex, half duplex and full-duplex using RS 232 C (9 pin) standard and bioscom function.
2. File transfer using RS 232C std. and bioscom function.
3. Simulation of different Framing methods. (Character count, starting and ending flag etc)
4. Implement error detection method – CRC
5. Implement error detection and correction method - Hamming code.
6. Implement a program for simulation of sliding window protocols.
7. Implementation of IEEE 802.3, 802.4 and 802.5 frame formats and conversion of 802.x frame into 802.y frame.
8. Implementation of Shortest path routing algorithm.
9. Implementation of Flow – based routing algorithm.
10. Given the IP address find out class, subnetmask, netid and hostid.

Text Books:

1. Data & Computer Communication (Unit 1) --William Stallings. (seventh edition) PHI publications.
2. Computer Networks (Unit 2, 3, 4, 5,6)--Andrew S. Tanenbaum (third edition) PHI publications.

Reference Books:

1. Computer Networks (Principles, Technologies and Protocols for network design) - Natalia Olifer, Victor Olifer (Wiley Publications)
2. Computer Networks, M. Dave, Cengage
3. An Engineering Approach to Computer Networking, Keshav, Pearson
4. An Integrated Approach to Computer Networks, Bhavneet Sidhu, Khanna Publications
5. Telecommunication Switching System and Networks, Viswanathan, PHI
6. Computer Networks - M. Dave, Cengage



**PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
S.Y.B.TECH (INFORMATION TECHNOLOGY)**

Semester-III

IT 214 - DIGITAL TECHNIQUES

Teaching Scheme

Lectures– 4 Hours/week, 4 Credits

Practical – 2 Hour/week, 1 Credit

Examination Scheme

ESE – 70 Marks

ISE – 30 Marks

ICA – 25 Marks

POE – 50 Marks

Introduction:

This course provides a thorough introduction to the Digital Logic Circuits and the basic building blocks used in digital systems. The course also introduces Hardware Description Language. It starts with a discussion of combinational logic circuits: logic gates, minimization techniques, arithmetic circuits, data processing circuits. The second part of the course deals with sequential circuits: flip-flops, registers and counters.

Course Prerequisite: Student has completed a Course in Basic Electronics and has an in-depth knowledge of electronic components and fundamentals of digital electronics.

Course Objectives:

1. To introduce various number systems and Boolean algebra.
2. To study basics of digital principles, digital logic circuits.
3. To develop designing and implementation skills for combinational and sequential logic circuits.
4. To design and simulate digital circuits using Hardware Description Language (HDL).

Course Outcomes:

At the end of course, students will be able to

1. Apply the fundamentals of various number systems, Boolean algebra to solve relevant problems.
2. Design and implement combinational and sequential logic circuits.
3. Simulate different logic circuits using Hardware Description Language (HDL).

SECTION I

UNIT-1	Digital Principles and Number Systems	10 Hrs.
	Introduction, Number Systems, Binary Number System, Signed Binary Numbers, Binary Arithmetic, 2's Complement Arithmetic, Octal Number System, Hexadecimal Number System, Codes. Introduction, Digital Signals, Basic Digital Circuits, NAND and NOR Operations, Exclusive-OR and Exclusive-NOR Operations, Boolean algebra, Examples of IC Gates.	
UNIT-2	Combinational Logic Design	10 Hrs.
	Introduction to Standard representation of Logical function, K-map representation, Simplification using k-map up to 4 variables, Minimization of logical function specified in minterms / maxterms, Don't care condition, AND / OR/ XOR function using NOR / NAND gates, Implementation of SOP & POS expression using NAND & NOR gate respectively, Arithmetic Circuits: Half adder, Full adder, Half subtractor, Full subtractor.	
UNIT-3	Combinational Logic Design using MSI Circuits	10 Hrs.
	Data selector / multiplexer, MUX as logic function, Decoder / Demultiplexes, application of MUX/ DEMUX using IC – 74151, 74154, 74148, problems based on MSI circuits, decoder / driver for 7 segment display using 7447. Digital Comparator, Parity Checker, Arithmetic Logic Unit.	

SECTION II

UNIT-4	Sequential Logic Circuit	10 Hrs.
	Introduction to flip-flops, S-R, J-K, D-Flip-flop & T-flip-flop, Excitation Tables for Flip flops, Basic register, Shift register, Asynchronous counters, Up-down counter, MOD counter, Introduction to synchronous counter.	
UNIT-5	Introduction to Verilog HDL	10 Hrs.
	Introduction to Verilog HDL, HDL Implementation Models: Dataflow Modeling, Behavioral Modeling, HDL Implementation of Arithmetic circuits: Half Adder, Full Adder, Half subtractor, Full subtractor. 4-bit ripple carry adder.	
UNIT-6	Digital Design with Verilog HDL	10 Hrs.
	HDL Implementation of Data Processing Circuits: 2:1, 4:1, 8:1 Multiplexer, BUS Representation in HDL, HDL Implementation of Flip-Flops, Register, Counters.	

Internal Continuous Assessment (ICA):

Student should perform 8 to 9 Experiments **using Trainer Kit / Bread Board** and **Synthesis tool: Xilinx ISE.**

1. a) Verification of truth table of basic and universal logic gates.
b) Implementation of Boolean functions using Basic and Universal Gates.
2. Implementation of reduced Boolean functions (K – map technique) from Assignment No 1 using basic and universal gates.
 - Implementation of half adder and full adder using Basic Gates.
 - a) Design and Implement a 4:1 multiplexer using 74153 IC.
Design and Implement a 8:1 multiplexer using 74151 IC.
 - Implementation of flip flops using NAND/NOR gates: S-R Flip flop, D flip flop.
 - Implementation of Mod 10 (Decade) Counter & Mod 5 counters using IC 7490.
 - a) Write Verilog code to realize all the logic gates, Simulate and verify it's working.
Design the Half subtractor and full subtractor using Verilog Code, Simulate and verify its working.
 - a) Design a 4:1 & 8:1 multiplexer circuit using Verilog Code, Simulate and verify its working.
Design a 1:8 Demultiplexer circuit using Verilog Code, Simulate and verify it's working.
 - Design the following flip-flops, SR, D, JK, T using Verilog Code, Simulate and verify its working.
 - Design 4 bit Synchronous and Asynchronous binary using Verilog Code, Simulate and verify its working.

Text Book:

- R. P. Jain: Modern Digital Electronics, 4th Edition, Tata McGraw-Hill Education, 2010.[Unit 1, 2, 3, 4]
- Donald P Leach, Albert Paul Malvino & Goutam Saha: Digital Principles and Applications, 7th Edition, Tata McGraw Hill, 2010.[Unit 5 & 6]

Reference Books:

- Samir Palnitkar: Verilog HDL: A Guide to Digital Design and Synthesis, Second Edition, Prentice Hall PTR, 2003.
- Stephen Brown, Zvonko Vranesic: Fundamentals of Digital Logic Design with VHDL, 2nd Edition, Tata McGraw Hill, 2005.
- R D Sudhaker Samuel: Illustrative Approach to Logic Design, Sanguine-Pearson, 2010.
- M Morris Mano: Digital Logic and Computer Design, 10th Edition, Pearson Education, 2008.
- Digital Electronics, Rishabh Anand, Khanna Publishing House



PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
S.Y.B.TECH (INFORMATION TECHNOLOGY)

Semester – III
IT 215 -COMPUTER GRAPHICS

Teaching Scheme

Lectures–3 Hrs./week, 3 credits

Practical–2Hrs./Week, 1 credit

Examination Scheme

ESE– 70 Marks

ISE– 30 Marks

ICA–25 Marks

Introduction:

This course introduces the basic of computer graphics and different basic graphics functions. It also develops ability for implementation of different algorithms. With this course student can acquire computer graphics techniques, its uses and implementation details.

Course Prerequisite: Knowledge of C Programming & Mathematics.

Course Objectives:

1. To introduce basic elements of computer graphics and graphic devices.
2. To study line, circle and polygon filling algorithms.
3. To get acquainted with 2D and 3D transformations.
4. To study line clipping algorithms.
5. To introduce hidden and visible surfaces and different types of curves.

Course Outcomes:

At the end of course, students will be able to

1. Implement algorithms to scan convert line, circle and polygon filling.
 2. Apply 2D and 3D transformations to graphic primitives.
 3. Demonstrate algorithms for line clipping, visible lines and surfaces algorithms.
 4. Incorporate built-in graphics functions for designing Animation.
-

SECTION I

UNIT-1	Basic Concepts & Devices	6 Hrs.
	Introduction to computer graphics, Application of Computer Graphics, pixel, frame buffer, resolution, aspect ratio, Video display devices: Refresh CRT, Raster scan display, Random scan display, color CRT monitors, Interactive devices: joysticks, touch panels, light pens.	
UNIT-2	Raster scan Graphics	8 Hrs.
	Line drawing algorithms: DDA, Bresenham's algorithm, Bresenham's Circle generation algorithm, RunLengthEncoding, Polygon filling: Scan converting polygon, Edge fill, Edge flag, Seed fill.	
UNIT-3	Geometric Transformations	9 Hrs.
	2D Transformation: Translation, Rotation, Reflection, Scaling, Shearing, Combined transformation, Rotation about an arbitrary point, Reflection through an arbitrary line. 3D Transformation: Scaling, Shearing, Rotation, Reflection, Translation, Multiple Transformation, Rotation about axis parallel to coordinate axis.	

SECTION II

UNIT-4	Clipping & Display File Compilation	7 Hrs.
	Sutherland-Cohen line clipping algorithm, Midpoint subdivision algorithm, Viewing transformation, Window transformation, segmented display file, Display file compilation.	
UNIT-5	Visible Lines & Visible Surfaces	8 Hrs.
	Hidden surfaces: introduction, back-face removal algorithm: Painter's algorithm, Warnock algorithm, Z-buffer. Antialiasing and antialiasing techniques, Half toning.	
UNIT-6	Plane curves & Space curves	7 Hrs.
	Introduction to curve generation, Curve representation, interpolation, Non parametric & parametric curves, Bezier curves, B-spline curves, Introduction to fractals, Fractal lines and surfaces.	

Internal continuous assessment (ICA):

Student should perform 8 to 10 experiments based on following guide lines.

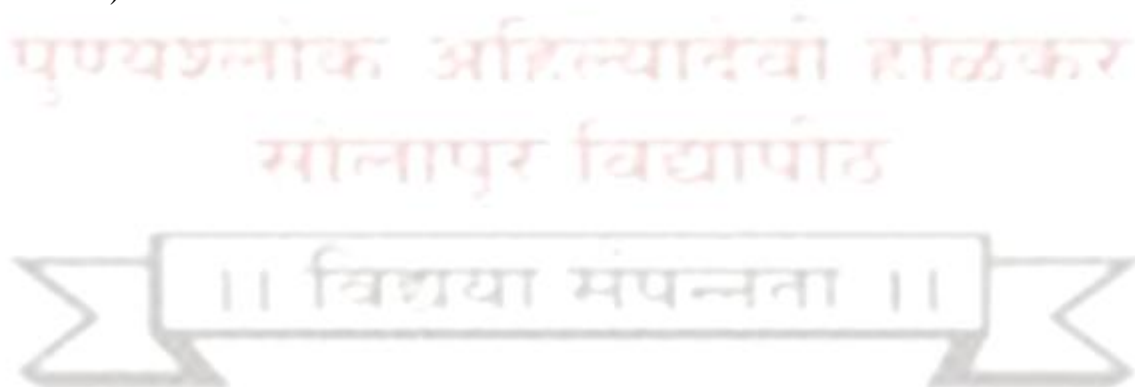
1. To Study Basic graphics functions.
2. Implementation of DDA line drawing algorithm.
3. Implementation of Bresenham's line drawing algorithm.
4. Implementation of Bresenham's Circle generation algorithm.
5. Implement Polygon filling algorithms.
6. Implement 2D transformation.
7. Implementation of 3D transformation.
8. Implement Sutherland – Cohen line clipping algorithm.
9. Implementation of Warnock algorithm.
10. Case study of OpenGL
11. Implement a small animation package.

Text Books:

3. Computer Graphics (**Chapter 1**) - Donald Hearn, Baker (second edition) PHI publications.
4. Procedural elements for Computer Graphics (**Chapter 2,4,5**) - David F. Rogers (second edition) Tata McGraw Hill publications.
5. Mathematical elements for Computer Graphics (**Chapter 3,6**) - Rogers, Adams (second edition) McGraw Hill Publishing Company.

Reference Books:

- Computer Graphics with virtual reality systems - Rajesh K. Maurya.
- Principles of Interactive Computer Graphics - William Newman, Sproull (second edition) McGraw-Hill Publication.





**PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
S.Y.B.TECH (INFORMATION TECHNOLOGY)**

Semester – III

IT 216 – ADVANCED C CONCEPTS

Teaching Scheme

Lectures– 2 Hrs./week, 2 Credits
Practical– 4 Hrs./Week, 2 Credits

Examination Scheme

POE– 50 Marks
ISE – 30 Marks
ICA–25 Marks

Introduction:

This course introduces study and implementation of advanced features of C Programming language. Storage classes, built-in library in C, string processing, pointer concepts. This course also introduces few data structure concepts such as Recursion, file concepts and sorting, searching and hashing techniques.

Course Prerequisite:

Student shall have basic C programming knowledge and background of mathematics.

Course Objectives:

1. To study storage classes and macros.
2. To apply recursion for different types of problems.
3. To study file and pointer basics and apply them to applications.
4. To study various searching and sorting algorithms.
5. To get acquainted with different hashing techniques.

Course Outcomes:

At the end of course, students will be able to

1. Use storage classes and preprocessors in C programs.
2. Develop solutions using recursion for given problems.
3. Apply pointers, string functions and file handling techniques for problem solving.
4. Implement searching, sorting and hashing techniques using C.

SECTION I

UNIT-1	C Preprocessor and Storage Classes	3 Hrs.
	Features of C Preprocessor, Macro Expansion - Macros with Arguments and Macros versus Functions, File Inclusion, Conditional Compilation - #if and #elif Directives; Storage Classes - Automatic, Register, Static, External storage classes.	
UNIT-2	Pointers	4 Hrs.
	Define Pointer, Pointer to Pointer, Pointers to Array, Array of Pointers, Pointer to String, Pointer to Structure, Pointer to Function, Memory Allocation - Static and Dynamic memory	
UNIT-3	Files	4 Hrs.
	Introduction, Streams and file types, File operations, Different File I/O Functions, other file functions, command line arguments.	
UNIT - 4	Recursion	4 Hrs.
	Definition and Process of Recursion in C, Types of Recursion, Comparison of Iterative and Recursion methods, How recursion works, Factorial, Fibonacci sequence, Towers of Hanoi, Advantages and Disadvantages of recursive techniques.	

SECTION II

UNIT - 5	String Processing	3 Hrs.
	user-defined functions to perform string operations – length, compare, concatenate, copy; array of strings	
UNIT - 6	Searching and Sorting	8 Hrs.
	Study and Implementation of Searching Algorithms- Linear search and Binary search. Study and Implementation of Sorting Algorithms- Bubble sort, Insertion sort, Merge sort, Quick sort, Selection sort, Shell sort and Radix sort.	
UNIT - 7	Hashing	4 Hrs.
	Different Hash Functions, choosing a hash function Collision Resolution by Open Addressing: Linear probing, quadratic probing, double hashing, Collision Resolution by Chaining.	

Internal Continuous Assessment (ICA):

ICA should be based on the following implementations using C programming language. Minimum 8 assignments should be conducted from following 11 assignments.

- Assignments based on C Preprocessor and Macro concepts.
 - Write a C program to define macro definitions for the following:
 - To test whether a character entered is a small case letter or not.
 - To test whether a character entered is an upper case letter or not.
 - To test whether a character is an alphabet or not. Make use of them macros you defined in (a) and (b) above.
 - To obtain the bigger of two numbers.
 - Write a C program to define macro definitions with arguments for calculation of area and perimeter of a triangle, a square and a circle. Store these macro definitions in a file called —areaperi.h. Include this file in your program, and call the macro definitions for calculating area and perimeter for different squares, triangles and circles.
- Assignments based on Storage classes in C.
 - Write a C program to demonstrate storage classes such as auto, register, static and extern.
- Assignments based on Pointer concepts.
 - Write a C program to demonstrate pointer concept.
 - Write a C program to demonstrate pointer to pointer concept.
 - Write a C program to demonstrate pointer to array concept.
 - Write a C program to demonstrate array of pointers concept.
- Assignments based on Pointer concepts.
 - Write a C program to demonstrate pointer to string concept.
 - Write a C program to demonstrate pointer to function concept.
 - Write a C program to demonstrate pointer to structure concept.
- Assignments based on Dynamic memory allocation concepts in C.
 - Write a C program to demonstrate use of calloc() function.
 - Write a C program to demonstrate use of malloc() function.
 - Write a C program to demonstrate use of realloc() function.
 - Write a C program to demonstrate use of free() function.
- Assignments based on File concepts in C.
 - Write a C program to open a given file, read a file and display file contents.
 - Write a C program to open a file and write to file.
- Assignments based on Recursion concept. (any 2)
 - Write a C program to calculate factorial of given number using recursive method.
 - Write a C program to display Fibonacci series using recursive method.
 - Write a C program to solve Towers of Hanoi problem using recursion method.
 - Write a C Program to find reverse of a number using recursion.
 - Write a C Program to convert decimal number to binary number using recursion.

Write a C Program to find power of a number using recursion.

- Assignments based on String operations.
 1. Write a C program to perform different string operations using string library functions such as string length, comparison, concatenation, and copy
 2. Write a C program to perform different string operations such as string length, comparison, concatenation, and copy using user-defined functions.
- Assignments based on searching concepts.
 1. Write a C program to implement Linear Search.
 2. Write a C program to implement Binary Search.
- Assignments based on sorting concepts. (any 2)
 1. Write a C program to implement Bubble sort.
 2. Write a C program to implement Insertion sort.
 3. Write a C program to implement Quick sort.
 4. Write a C program to implement Merge sort.
 5. Write a C program to implement Selection sort.
 6. Write a C program to implement Shellsort.
- Assignments based on Hashing concepts using array (any 2)
 1. Write a C program to implement Hashing concept.
 2. Write a C program to demonstrate collision in hashing technique.
 3. Write a C program to demonstrate collision resolution using Open Addressing method i.e. Linear probing

Text Books:

- Let Us C by Yashvant Kanetkar [Unit-1, 3, 5]
- Pointers in C by Yashvant Kanetkar [Unit- 2]
- Data Structure using C and C++ second edition by Yedidyah Langram , Moshe J, Augensteen, Aason M. Tanenbaum [Unit- 4, 7]
- C with Data Structures by Dr. R. Nageswara Rao [Unit-1, 2, 3, 5, 6]

Reference Books:

- Data Structures through C in Depth, S.K. Srivastava, Deepali Srivastava, BPB Publications
- Data Structures Using C & C++ by Rajesh Shukla.
- Data Structures A Pseudocode Approach with C by Richard F. Gilberg & Behrouz A. Forouzan
- Data Structures through C, Yashwant Kanetkar, BPB Publications



**PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY,
SOLAPUR**

S.Y.B.TECH (INFORMATION TECHNOLOGY)

Semester-IV

IT 221 APPLIED MATHEMATIC-II

Teaching Scheme

Lectures – 3 Hours/week, 3 Credits

Tutorial – 1 Hour/week, 1 Credit

Examination Scheme

ESE – 70 Marks

ISE – 30 Marks

ICA- 25 Marks

Introduction:

This course introduces numerical methods (Unlike analytical methods) to solve algebraic and transcendental equations, simultaneous systems of linear equations and numerical integrations. This course also introduces the fuzzy set theory in brief which deals with characterizing the concept of uncertainty and its relationship to the increasingly important concept of information and complexity. This course also introduces simplex method to solve LPP and assignment problems.

Course Prerequisite:

Student shall have knowledge of basic notions of classical set theory and probability theory. Student shall have to be familiar with some analytical method for solving equations, simultaneous equations & analytical methods to solve definite integrations.

Course Objectives:

- 1) To make students familiar with the uncertainty.
- 2) To give students comprehensive coverage of operations on fuzzy sets.
- 3) To make student use of numerical methods for the problems that cannot be solved analytically.
- 4) To enable students to solve Linear Programming Problems and Assignment Problems.

Course Outcomes:

At the end of course, students will be able to

- 1) Solve non linear algebraic and transcendental equations.
- 2) Solve simultaneous linear and non linear equations.
- 3) Apply numerical methods to evaluate definite integrals.
- 4) Apply knowledge of basics of fuzzy set theory to solve the problems.
- 5) Solve the fuzzy equations
- 6) Solve a particular kind of problems arises in day to day life using simplex method and Assignment Problems.

SECTION I

Unit 1: Solution of algebraic and transcendental equation

No of lectures – 07

Basic properties of equations, False position method, Newton-Raphson method, Multiple roots, Newton's iterative formula for obtaining square root, system of non-linear equations by Newton Raphson method.

Unit 2: Solution of linear simultaneous equations

No of lectures – 07

Direct methods – Gauss Elimination method, Gauss Jordan methods, Method of factorization Iterative methods – Jacobi's method, Gauss-seidal method, power method to find eigen value and eigen vector.

Unit 3: Numerical Integration

No of lectures – 07

Numerical integration using Newton's cote's formulae – Trapezoidal rule, simpson's 1/3rd rule, Simpson's 3/8th rule, Weddel's rule, Romberg integrations, Double integrations.

SECTION II

Unit 4: Classical (Crisp) sets to fuzzy sets:

No of lectures – 06

Crisp sets, Basic types of fuzzy sets, Basic concepts of fuzzy sets, fuzzy sets vs Crisp sets: Additional properties of α -cuts, representation of fuzzy sets and extension principle of fuzzy sets.

Unit 5: Fuzzy arithmetic.

No of lectures – 08

Fuzzy number, arithmetic operations on intervals, arithmetic operations on fuzzy numbers, fuzzy equations, lattice of fuzzy numbers.

Unit 6: LPP and Assignment Problems.

No of lectures – 07

Introduction of LPP, Simplex method for LPP, Assignment problem: introduction mathematical formulation of Assignment Problem, Hungarian method to solve Assignment Problem.

Internal Continuous Assessment (ICA):

ICA shall consist of minimum six to eight assignments based on entire curriculum

Text Books:

- 1) B.S. Grewal, Numerical methods, Khanna publication, New Delhi.
- 2) George j Klir and Bo Yuan, Fuzzy sets and Fuzzy logic – PHI India.
- 3) Fundamental of statistics, S.C. Gupta, Himalaya house publication.

Reference Books:

- 1) George J. Klir and Tina A. Folger, Fuzzy sets, uncertainty and information, PHI India.
- 2) Robert J. Schilling, Sandra L. Harris, Applied Numerical methods for Engineers.
- 3) M.K. Jain, S.R.K. Iyengar, R.K. Jain, Numerical methods for scientific and engineering computations – New Age International ltd.
- 4) Pundir & Pundir, Fuzzy sets and their applications – Pragati Publications.



**PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY,
SOLAPUR**

S.Y.B.TECH (INFORMATION TECHNOLOGY)

Semester-IV

IT 222 - THEORY OF COMPUTATION

Teaching Scheme

Lectures– 4 Hours/week, 4 credits

Tutorial – 1 Hour/week, 1 credit

Examination Scheme

ESE – 70 Marks

ISE – 30 Marks

ICA - 25 Marks

Introduction:

Theory of computation lays a strong foundation for a lot of abstract areas of computer science. TOC teaches you about the elementary ways in which a computer can be made to think. Any algorithm can be expressed in the form of a finite state machine and can serve as a really helpful visual representation of the same. Sometimes, the finite state machines are easier to understand thus helping the cause furthermore.

Prerequisite: Students should have prior knowledge of Discrete Mathematical Structure

Course Objectives:

1. To study computational principles to build regular expressions for a given regular language.
2. To get acquainted with different types of automata.
3. To gain in-depth knowledge about regular and non-regular languages.
4. To study how to detect and remove ambiguity in a grammar.
5. To become familiar with different types of abstract machines.

Course Outcomes:

At the end of course, students will be able to

1. Build regular expression for a given language.
2. Design an automata for given formal languages.
3. Classify a given language into regular and non regular language.
4. Detect ambiguity in grammar and remove it.
5. Design abstract machines for a given language.

SECTION-I

UNIT-1	Regular Expressions	8 Hrs.
	Regular expressions & corresponding regular languages, examples and applications, unions, intersection & complements of regular languages	
UNIT-2	Finite Automata	8 Hrs.
	Finite automata definition and representation, Non-deterministic F.A., NFA with \wedge transitions, Equivalence of DFA & NFA	
UNIT-3	Kleen's Theorem	8 Hrs.
	Statements & proofs, minimizing number of states in an FA, Basics of Moore and Mealy Machines	
UNIT – 4	Grammars & Languages	8 Hrs.
	Definition and types of grammars and languages, derivation trees and ambiguity, CNF notations, Union, Concatenation and *'s of CFLs, Eliminating production and unit productions from a CFG, Eliminating useless variables from a Context Free Grammar	

SECTION-II

UNIT – 5	Push down Automata	7 Hrs.
	Definition, deterministic PDA & types of acceptance, equivalence of CFGs & PDAs.	
UNIT – 6	CFL's & Non CFL's	6 Hrs.
	Pumping Lemma & examples, intersection and complements.	
UNIT – 7	Turing machines	10 Hrs.
	Models of computation, definition of TM as language Acceptors, Combining Turing machines, computing a function with a TM.	
UNIT - 8	Variations in TM	5 Hrs.
	TMs with doubly infinite tapes, Multitape, Non-deterministic TM and universal TM.	

Internal Continuous Assessment (ICA) :

Students should solve assignments based on the topics below:

- Regular Expression & Corresponding Languages
- Union, Intersection & Complements of Regular languages
- Design & Simulation of Simple Finite Automata
- Nondeterministic Finite Automata & NFA with λ transitions ,Conversion of NFA to DFA
- Draw NFA using Kleens theorem
- DFA minimization
- Grammar ,Removing ambiguity from a grammar ,Conversion to BNF & CNF form
Push Down Automata
- Pumping Lemma & Examples for regular sets & regular languages
- Turing Machine

Books:

- Introduction to languages & theory of computation -- John C. Martin (MGH)
- Formal Languages & Automata Theory-- Basavraj S. Anami, Karibasappa K.G., Wiley Precise Textbook-Wiley India

References:

- Theory of Computation—Rajesh K Shukla (CENGAGE Learning)
- Introduction to Automata theory, languages and computations – John E. Hopcraft, Rajeev Motwani, Jeffrey D. Ullman (Pearson Edition).
- Discrete mathematical structures with applications to Computer science -- J.P.Tremblay & R.Manohar (MGH)
- Theory of Computer Science: Automata, Languages and Computation, Mishra, PHI
- Theory of Computation, RB Patel & Prem Nath, Khanna Publications





PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
S.Y.B.TECH (INFORMATION TECHNOLOGY)

Semester – IV

IT 223 – MICROPROCESSORS

Teaching Scheme

Lectures– 3 Hrs./week, 3 Credits

Practical– 2Hrs./week, 1 Credit

Examination Scheme

ESE– 70 Marks

ISE– 30 Marks

ICA– 25 Marks

POE–50 Marks

Introduction:

This course introduces to develop an in-depth understanding of the operation of microprocessor, machine language programming and interfacing techniques.

Prerequisite: Knowledge of Digital Techniques and Basic Electrical Engineering

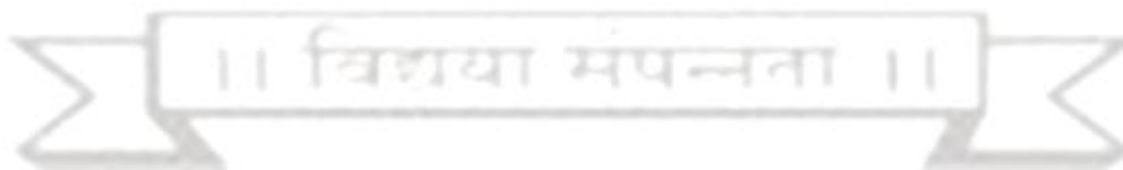
Course Objectives

1. To introduce 8085, 8086, 80286, 80386 and 80486 microprocessor architectures and their functionalities.
2. To get acquainted with programming model of 8086 microprocessor and develop assembly language programs
3. To study the interfacing between 8086 microprocessor and various peripherals.
4. To get acquainted with system bus architecture of 8086 microprocessor

Course Outcomes:

At the end of course, students will be able to

1. Demonstrate the architecture and its functionality for various Intel family microprocessors.
2. Implement assembly language programs for 8086 microprocessor.
3. Illustrate interfacing of various peripheral devices with 8086 microprocessor.
4. Describe the use of system bus in case of single and multiprocessor environments.

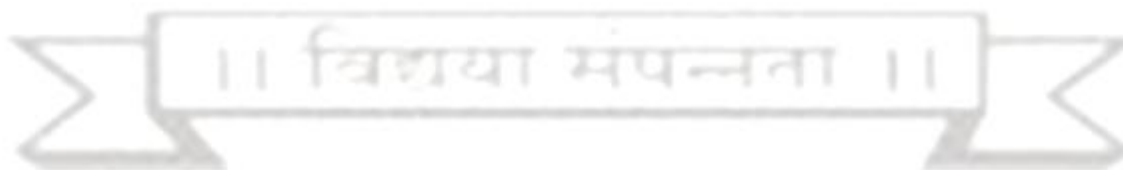


SECTION-I

UNIT-1	Introduction to Microprocessor	4 Hrs.
	Introduction to microprocessor, Features and pin diagram of 8085, 8085 MPU architecture,	
UNIT-2	8086 Microprocessor	6 Hrs.
	8086 Architecture, Internal Operation, Machine Language Instructions: Addressing Modes, Instruction Execution Timing	
UNIT-3	Assembly Language Programming	6 Hrs.
	Assembler Instruction Format, 8086 instruction set, Directives	
UNIT -4	System Bus Architecture	4 Hrs.
	Introduction , Basic 8086 Configurations : Minimum Mode, Maximum Mode , System Bus Timing	

SECTION-II

UNIT-5	Interrupts	6 Hrs.
	Introduction, Types of 8086 Interrupts, Interrupt and Interrupt service Routine, Maskable and Non-maskable Interrupt, Programmable Interrupt Controller (8259), Programmable Features and Block Diagram of DMA Controller 8257, Operating modes of 8257.	
UNIT-6	Programmable peripheral Interface	6 Hrs.
	Programmable peripheral Interface 8255, Modes of Operation	
UNIT-7	Programmable Communication Interface and 8087 NDP	6 Hrs.
	8087 numeric data processor. NDP data types, processor architecture	
UNIT-7	The 80286, 80386 and 80486 Microprocessor	7 Hrs.
	Introduction to 80286 Microprocessor, Architecture, Real Address Mode Operation, Protected Mode operation, 80386 Architecture, 80486 Architecture	



Internal Continuous Assessment (ICA) :

Student should perform 8 to 10 experiments using TASM/MASM based on following guide lines.

1. Addition and subtraction of two 16 bit numbers
2. Addition and subtraction of two 32 bit numbers
3. 16 bit multiplication of unsigned numbers.
4. 8 bit division of unsigned numbers
5. Find factorial of number
6. Generate a Fibonacci series.
7. Program for block transfer
8. Program to arrange numbers in ascending and descending order
9. Program to find Largest No. in a block of data.
10. Program to display the string.
11. Program to implement key board sensing using 8255
12. Implementation of 7-segment display using 8255

Text Books:

- Microprocessor Architecture, Programming, and Applications with the 8085 - Ramesh Gaonkar, Fifth edition. (UNIT -1)
- Lise & Gibson, —Microcomputer System 8086 / 8088| PHI, 2nd Edition. (Unit 2, 3, 4, 7)
- D. V. Hall, —Microprocessor and Interfacing Programming & Hardware| TMH – 2nd Edition- (Unit 8)
- A K Ray & K M Bhurchandi —Advanced Microprocessors and Peripherals|. 2nd Edition (Unit 5,6)

Reference Books:

1. Barry B. Brey, —The Intel Microprocessors: Architecture, Programming & Interfacing| PHI, 6th Edition.
2. Uffenback, —The 8086 Family Design| PHI, 2nd Edition.
3. Advanced Microprocessors, AK Gautam, Khanna Publishing House





PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
S.Y.B.TECH (INFORMATION TECHNOLOGY)

Semester – IV
IT 224 - DATA STRUCTURES

Teaching Scheme

Lectures– 3 Hrs./week, 3 Credits
Practical– 4 Hrs./Week, 2 Credits

Examination Scheme

ESE– 70 Marks
ISE– 30 Marks
ICA–25 Marks
POE– 50 Marks

Introduction:

This course introduces various linear data structures like stack, queue, linked list and non-linear data structures like various kinds of trees and graphs. Course includes implementation of various operations of these data structures and some applications.

Course prerequisites:

This course requires prior knowledge of advanced C.

Course Objectives:

1. To study various data structures.
2. To develop programming skills to implement and analyze linear and non linear data structures.
3. To identify and apply suitable data structure for solving a given problem.

Course Outcomes:

At the end of course, students will be able to

1. Describe and distinguish linear and non-linear data structures.
2. Implement various data structures and perform its various operations using C
3. Select appropriate data structures for solving a given problem using C programming.

SECTION I

UNIT - 1	Introduction to Data Structures & Stack	5 Hrs.
	What is Data Structure, types of data structures – static, dynamic, primitive, non-primitive, linear, non-linear Stack: Definition, representation, Operations, Implementation and applications like conversion of polish notations, evaluation of postfix expressions.	
UNIT - 2	Queues	4 Hrs.
	Definition, representation, Operations, Implementation: Linear Queue, Circular Queue, Priority Queue.	
UNIT - 3	Lists	6 Hrs.
	Definition, representation, Operations, Types of Lists: Singly Linked	

	list, Doubly Linked list, Circular Linked list, Stack using linked list, Queue using Linked list, Application of Linked list: Addition and Subtraction of two polynomials	
UNIT - 4	Trees	8 Hrs.
	Definition, Traversal, Linked implementation, Operations on: Binary trees and Binary Search Trees, Introduction to Threaded Binary trees	

SECTION –II

UNIT - 5	Multiway Trees	7 Hrs.
	Multiway search Trees, Balanced Multiway Trees, Traversing a Multiway Tree, Insertion in Multiway Tree: B Trees, B+ Trees	
UNIT - 6	Height Balance Trees	7 Hrs.
	AVL Trees: Definition, Height of an AVL Tree, Insertion, Deletion of node in AVL Trees, Single and Double rotation of AVL Trees.	
UNIT - 7	Graphs	8 Hrs.
	Definition, Undirected and Directed Graphs, Graph Terminologies, Computer Representation of Graphs, Graph Traversal methods: Depth First and Breadth First Search, Application: Shortest Path using Dijkstra's algorithm.	

Internal Continuous Assessment (ICA) :

List of Assignments: (minimum 18 assignments)

- Implementation of Stack using array
- Implementation of Stack using structure.
- Conversion of infix expression to postfix expression using Stack.
- Evaluation of postfix expression using Stack.
- Implementation of Linear Queue using array.
- Implementation of Circular Queue using structures.
- Implementation of Priority Queue.
- Implementation of Singly Linked List and operations on it.
- Implementation of Doubly Linked List and operations on it.
- Implementation of Singly Circular Linked List.
- Implementation of doubly Circular Linked List.
- Implementation of Stack using Linked List.
- Implementation of Queue using Linked List.
- Implementation of polynomial addition using Linked List.
- Implementation of Binary Search Tree and its traversal (In order, Preorder, Post order).
- Implementation of Binary Search Tree and deletion of nodes in it.
- Implementation of graph using adjacency matrix.
- Implementation of graph using adjacency list.

- Implementation of Depth First Traversal of Graph.
- Implementation of Breadth First Traversal of Graph.
- Creation of B Tree of order 3 or 5 by inserting nodes in it using some example.
- Creation of B+ Tree of order 3 or 5 by inserting nodes in it using some example.
- Creation of of AVL Tree by performing different rotations.

Text Books:

- Data Structure and Program Design in C by Robert Kruse/C.L.Tonda/BruceLeung second edition, Pearson Education, Prentice Hall.
- Data Structures: A Pseudo Approach with C. by Richard.F.Gilberg& Behrouz .A. Forouzan, second edition, Cengage Learning
- Data Structure using C and C++ by Rajesh.K.Shukla, Wiley Publication

Reference Books:

- Data Structures using C and C++, second edition by YedidyahLangram, Moshe J, Augenstein, Aason .M. Tanenbaum.
- Data Structures and Algorithms by Prof Maria .S. Rukadikar, Shroff Publications.
- Data Structures Through C in Depth by S.K. Shrivastava, Depali Shrivastava, BPB Publications
- 4 Fundamentals of Data Structures, Sartaj Sahni, University Press
- 5 Data Structures, RS Salaria, Khanna Publishing House
- 6 Data Structures through C, Yashwant Kanetkar, BPB Publications
- 7 Expert Data Structures with C++, RB Patel, Khanna Publications

पुण्यश्लोक अहिल्यादेवी संज्ञक
सोलापूर विद्यापीठ

॥ विशया मंपन्नता ॥



**PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
S.Y.B.TECH (INFORMATION TECHNOLOGY)**

Semester-IV

IT 225 – COMPUTER NETWORKS

Teaching Scheme

Lectures – 3 Hours/week, 3 Credits

Practical – 2 Hour/week, 1 Credits

Examination Scheme

ESE – 70Marks

ISE – 30Marks

ICA - 25 Marks

Introduction:

This course introduces TCP/IP protocol in detail and it also covers the IPv4 Addressing, Socket Programming, Transport layer and Application layer protocols.

Course Prerequisite: Student should have the knowledge of Data communication fundamentals and types of Computer Networks

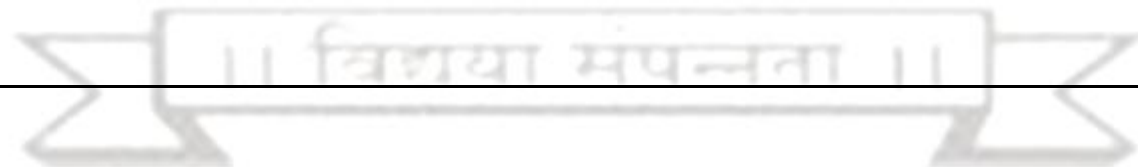
Course Objectives:

1. To study various classes of network addressing for designing a LAN .
2. To introduce client-server paradigm for socket interface using transport layer protocols.
3. To provide in-depth knowledge of application layer protocols

Course Outcomes:

At the end of course, students will be able to

1. Describe different classes of addressing and select appropriate while designing a LAN .
2. Implement client-server paradigm for socket interfaces using transport layer protocols.
3. Select & use appropriate application layer protocols for given problem.



SECTION-I

UNIT-1	IP Protocol	7 Hrs.
	Internet Protocol: Introduction, IP Datagram, fragmentation, Addressing: Physical , Logical, Port & Application Specific Addresses. Introduction to IPv4 Addresses: Classful addressing, Classless addressing, Special addresses, NAT	
UNIT-2	Transport Layer	8 Hrs.
	UDP: Introduction, User Datagram, UDP Services, UDP Applications. TCP: TCP Services, TCP Features, Segment, A TCP Connection, State Transition Diagram, Window in TCP, Flow Control, Error Control, Congestion Control, TCP Timers. SCTP: Introduction, SCTP Services, SCTP Features, Packet Format	
UNIT-3	Client Server Model and Socket Interface	8 Hrs.
	Client Server Paradigm: Server, Client, Concurrency, Concurrency in Clients, Concurrency in Servers, Socket, Byte Ordering Functions. Socket System Calls, Connectionless Iterative Server, UDP Client Server Programs, and Connection- oriented Concurrent Server.	

SECTION -II

UNIT- 4	Host Configuration & Domain Name System	7 Hrs.
	Host Configuration: BOOTP Operation, Packet format, DHCP: Introduction, DHCP Operation and Configuration. Domain Name System: Need for DNS, Name Space, DNS in the Internet, Resolution, DNS Messages, Types of Records.	
UNIT- 5	Remote Login and TELNET	7 Hrs.
	TELNET Concept, Time-Sharing Environment, Network Virtual Terminal, Embedding, Options, Symmetry, Sub option Negotiation, Controlling the Server, Out-of-Band Signalling, Escape Character, Mode of Operation, User Interface. SSH: Components, Port Forwarding, Format of SSH Packets.	
UNIT- 6	File Transfer and Electronic Mail	8 Hrs.
	FTP: Introduction, control & data connections, Communication over data and control connection, Command Processing TFTP: Messages, Connection, Data Transfer, UDP Ports, TFTP Applications. Electronic Mail: Architecture, User Agent, Message Transfer Agent, SMTP, Message Access Agent: POP and IMAP	

Internal Continuous Assessment (ICA):

Students should perform minimum 8 experiments based on the following guidelines and preferably conducted on Unix / Linux platform using C language.

1. Configuration of Network-Assigning IP Address, Subnet-Mask, Default Gateway, DNS Server Addresses & Testing Basic Connectivity.
2. Connectionless Iterative Server: C Implementation of Client-Server Programs Using Iterative UDP Server.
 - Connection-oriented Iterative Server: C Implementation of Client-Server Programs Using Iterative TCP Server.
 - Connection-oriented Concurrent Server: C Implementation of Client-Server Programs Using Concurrent TCP Server.
 - Implementation of Simple Network Chatting Application.
 - Remote Login: TELNET
 4. Log on to a remote computer from client using TELNET.
 5. After logging on executes few commands at remote server from client. For example user wants a server to display a file (hello.txt) on a remote server then he/she types:
cat hello.txt.
 6. Log on to a remote computer from client using TELNET and Putty terminal emulator. After logging on execute few commands. Here Client and Server are on heterogeneous systems, for example client is on windows and server is on Linux.
 - Remote Login: SSH
 4. Log on to a remote computer from client using SSH.
 5. After logging on executes few commands at remote server from client. For example user wants a server to display a file (hello.txt) on a remote server then he/she types:
cat hello.txt.
 6. Log on to a remote computer from client using SSH and Putty terminal emulator. After logging on execute few commands. Here Client and Server are on heterogeneous systems for example client is on windows and server is on Linux.
 - Installation and configuration of DHCP
 - Installation and configuration of FTP.

Text Books :

- TCP/IP Protocol Suite: Behrouz A. Forouzan (Fourth Edition) (Unit 1,2,3,5,6)
- TCP/IP Protocol Suite: Behrouz A. Forouzan (Third Edition) (Unit 4)
- TCP/IP Protocol Suite: Behrouz A. Forouzan (Second Edition) (Unit 3)
- Computer Networking: A Top-Down Approach Featuring the Internet, International Edition: James F. Kurose and Keith W. Ross

Reference Books :

- Internetworking with TCP/IP Vol III. Client-Server Programming & Applications: Douglas E. Comer
- Data and Computer Communications: William Stallings
- Data Communication and Networking: Behrouz A. Forouzan
- 4. Computer Networks, M. Dave, Cengage
- 5. An Engineering Approach to Computer Networking, Keshav, Pearson
- 6. An Integrated Approach to Computer Networks, Bhavneet Sidhu, Khanna Publications
- 7. Telecommunication Switching System and Networks, Viswanathan, PHI





**PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY,
SOLAPUR**

S.Y.B.TECH (INFORMATION TECHNOLOGY)

Semester – IV

IT 226 - OBJECT ORIENTED PROGRAMMING THROUGH C++

Teaching Scheme

Theory – 2 Hrs./Week, 2 credits
Practical–2 Hrs./Week, 1 credit

Examination Scheme

ICA – 25 Marks
ISE – 25 Marks
POE – 50 Marks

Introduction:

OOP provides a clear modular structure for program. OOP provides a good framework for code libraries where supplied software components can be easily adapted and modified by the programmer. C++ has a longer history with game development in general. It's a device level programming, Multi-paradigm, compiled, general-purpose programming language. It is regarded as a "middle-level" language, as it comprises a combination of both high-level and low-level language features.

Course Prerequisite:

Student should have programming language fundamentals.

Course Objectives:

1. To study fundamental concepts and features of Object Oriented Programming.
2. To implement File Handling Operations in C++
3. To implement C++ program using templates.
4. To develop solutions using Object Oriented Programming.

Course Outcomes:

At the end of course, students will be able to

1. Implement Object Oriented Programming constructs & features using C++.
2. Implement File Handling Operations in C++.
3. Implement C++ program using templates.
4. Develop solutions using Object Oriented Programming.

SECTION – I

UNIT - 1	OOP Concepts	3 Hrs.
	Basic Concepts, Features, C++ Programming Basics, Object and Class, Array of objects, Constructors & types of constructors, Destructor.	
UNIT - 2	Functions	4 Hrs
	Default & Reference arguments, Function Overloading, Inline functions, Return by reference, Friend Functions and Static Functions.	
UNIT - 3	Operator Overloading	4 Hrs
	Definition, Overloading unary and binary operators, Overloading Extraction and Insertion Operators, Data Conversions	
UNIT - 4	Inheritance	4 Hrs
	Derived class and base class, Types of inheritance, Derived class constructors, Overriding Member Functions, Nesting of Classes.	

SECTION II

UNIT – 5	Pointers and Virtual Functions	5 Hrs
	Memory Management Operators - new and delete, Pointers to objects, this pointer, Pointers to Derived classes, Virtual Function, Late Binding, Pure Virtual Functions, and Abstract Base Classes	
UNIT – 6	Manipulators and File Handling	4 Hrs
	C++ Streams, C++ stream classes, Managing Output with Manipulator, File Stream Classes, Working with File, Object I/O, Command Line Arguments.	
UNIT – 7	Template and Exception Handling	4 Hrs.
	Class Templates, Function template, Exception handling, Throwing Mechanism, Catching Mechanism, Rethrowing an Exception.	
UNIT - 8	Introduction to Standard Template Library	2 Hrs.
	Components of STL, Containers, Algorithms, Iterators, Application Container classes.	

Internal Continuous Assessment (ICA) :

Student should implement minimum 8 assignments from the following.

- Assignment on Class, Constructor, Destructor
- Assignment on function Overloading, Constructor Overloading
- Assignment on Operator Overloading
- Assignment on Multiple Inheritance, Multilevel Inheritance
- Assignment on Static Variables & Function in class
- Assignment on Virtual Function, Virtual Class
- Assignment on Function template, Class template
- Assignment on Friend class and function,
- Assignment on File handling
- Assignment on Exception handling.
- An individual student has to submit a project using C++ language as a part of ICA.

Text Books:

1. Object Oriented Programming with C++ - E. Balagurusamy (McGraw-Hill)
2. Programming with C++ - D. Ravichandran (TMGH)

Reference Books:

- C++ programming language - Bjarne Stroustrup (AT & T)
- Object oriented programming in Turbo C++ - Robert Lafore (Galgotia)
- Mastering Object-Oriented Programming with C++, R.S. Salaria, Khanna Book Publishing, N. Delhi

