PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR

FACULTY OF SCIENCE & TECHNOLOGY

COMPUTER SCIENCE & ENGINEERING

Syllabus Structure for

First Year (All Branches) w.e.f. Academic Year 2018-19

Second Year B. Tech. (Computer Science & Engineering) w.e.f. Academic Year 2019-20

Third Year B. Tech. (Computer Science & Engineering) w.e.f. Academic Year 2020-21

Choice Based Credit System

Programme Educational Objectives and Outcomes

A. Program Educational Objectives

1. Graduate will exhibit strong fundamental knowledge and technical skills in the field of Computer Science & Engineering to pursue successful professional career, higher studies and research.
2. Graduate will exhibit capabilities to understand and resolve various societal issues through their problem solving skills.
3. Graduate will be sensitive to ethical, societal and environmental issues as a software engineering professional and be committed to life-long learning.

B. Program Outcomes

Engineering Graduate will be able to –

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **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.
3. **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.
4. **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.
5. **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.
6. **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.

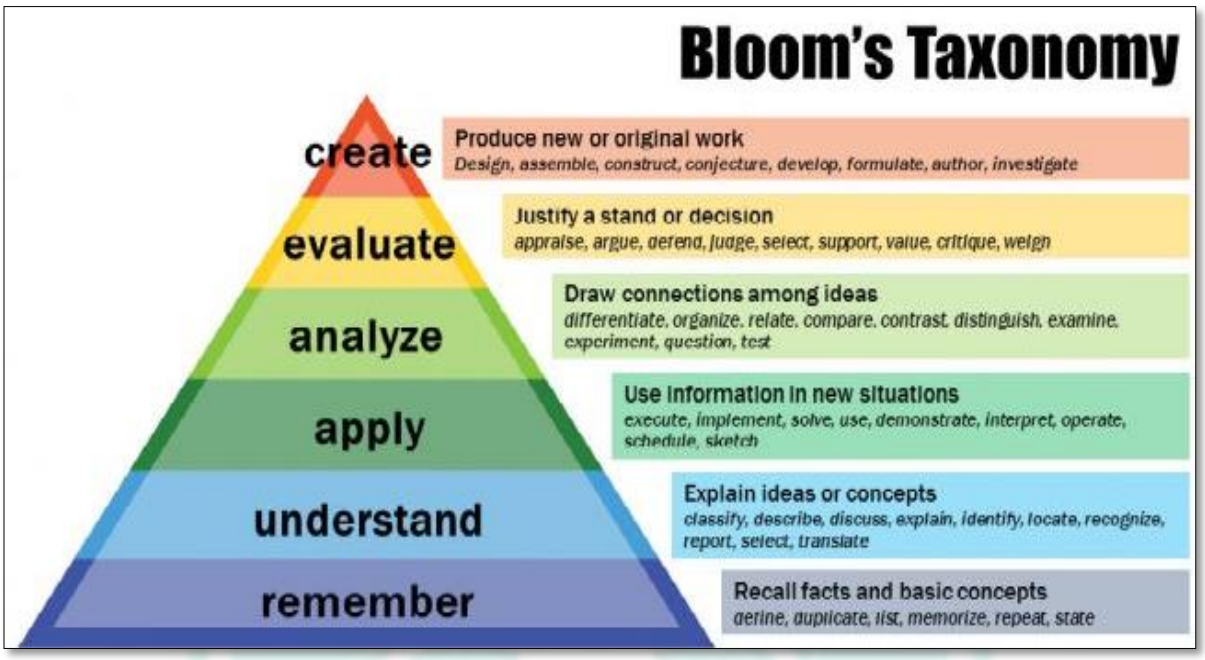
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **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.
11. **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.

C. Program Specific Outcomes (PSOs)

1. Apply the principles of computational mathematics, computer systems and programming paradigms to solve computational problems.

2. Design and develop application software with functionalities applicable for desktop, web and mobile applications with due consideration of system software constraints.

3. Apply software engineering methods, cutting edge technologie and ICT, using appropriate tools and FOSS alternatives for designing ,developing & testing application software



- 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.

An induction program for the new UG entrant students is proposed at the commencement ofthe 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, SOAPUR
FACULTY OF SCIENCE & TECHNOLOGY
Credit System structure of Second Year B.Tech. (CSE) wef. 2019-2020

Semester – I

Course Code	Theory Course Name	Hrs./Week			Credits	Examination Scheme			
		L	T	P		ISE	ESE	ICA	Total
CS211	Applied Mathematics-I	3	1	--	4	30	70	25	125
CS212	Discrete Mathematical Structures	3	1	--	4	30	70	25	125
CS213	Data Communication	3	--	--	3	30	70	--	100
CS214	Digital Techniques	4	--	--	4	30	70	--	100
CS215	Computer Graphics	3	--	--	3	30	70	--	100
CS216	Advanced C Concepts	2	--	--	2	25	--	--	25
	Sub Total	18	2	--	20	175	350	50	575
Course Code	**Laboratory Course Name**								
							ESE / **POE**	**ICA**	
CS213	Data Communication	--	--	2	1	--	50	25	75
CS214	Digital Techniques	--	--	2	1	--	50	25	75
CS215	Computer Graphics	--	--	2	1	--	--	25	25
CS216	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**
ENV21	**Environmental Studies**	1	--	--	--	--	--	--	--

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

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOAPUR
FACULTY OF SCIENCE & TECHNOLOGY
Credit System structure of Second Year B.Tech. (CSE) wef. 2019-2020
Semester – II

Course Code	Theory Course Name	Hrs./Week			Credits	Examination Scheme			
		L	T	P		ISE	ESE	ICA	Total
CS221	Applied Mathematics-II	3	1	--	4	30	70	25	125
CS222	Theory of Computation	4	1	--	5	30	70	25	125
CS223	Microprocessors	3	--	--	3	30	70	--	100
CS224	Data Structures	3	--	--	3	30	70	--	100
CS225	Computer Networks	3	--	--	3	30	70	--	100
CS226	Object Oriented Programming through C++	2	--	--	2	25	--	--	25
	Sub Total	**18**	**2**	**--**	**20**	**175**	**350**	**50**	**575**
Course Code	**Laboratory Course Name**								
							ESE / **POE**	**ICA**	
CS223	Microprocessors	--	--	2	1	--	50	25	75
CS224	Data Structures	--	--	4	2	--	50	25	75
CS225	Computer Networks	--	--	2	1	--	--	25	25
CS226	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**	**150**	**825**
ENV22	**Environmental Studies**	1	--	--	--				

- *Abbreviations: L - Lectures, P –Practical, T - Tutorial, ISE - In Semester Exam., ESE- End Semester Exam, ICA - Internal Continuous Assessment, ISE - Internal Tests, 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. 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 SCIENCE & TECHNOLOGY
Credit System Structure of Third Year B.Tech. (CSE) wef. 2020-2021
Semester – I

Course Code	*Theory Course Name*	*Hrs./Week*			*Credits*		*Examination Scheme*		
		L	*T*	*P*		*ISE*	*ESE*	*ICA*	*Total*
CS311	System Programming	3	--	--	3	30	70	--	100
CS312	Operating Systems	3	--	--	3	30	70	--	100
CS313	Software Engineering	3	--	--	3	30	70	--	100
CS314	$ Database Engineering	4	--	--	4	30	70	--	100
CS315	Design and Analysis of Algorithm	3	--	--	3	30	70	--	100
CS316	Python Programming	2	--	--	2	25	--	--	25
CS317	Java Programming	2	--	--	2	25	--	--	25
SL31	Self Learning Module I (HSS)	--	--	--	2	--	50	--	50
	Sub Total	20	--	--	22	200	400	--	600
Course Code	**Laboratory Course Name**								
							ESE	**ICA**	
							POE	**OE**	
CS311	System Programming	--	--	2	1		--	--	25	25
CS313	Database Engineering	--	--	2	1		50	--	25	75
CS314	Design and Analysis of Algorithm	--	--	2	1		--	--	25	25
CS316	Python Programming	--	--	2	1		50	--	25	75
CS317	Java Programming	--	--	2	1		50	--	25	75
	Sub Total	--	--	10	5		150	--	125	275
	Grand Total	**20**	**--**	**10**	**27**	**200**	**550**		**125**	**875**

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

$ - The theory courses for Computer Sci. and Engg. and Information Technology are same, therefore paper for ESE will be common to both.

Note :

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 vacation/s 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. 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.

4. **Self-Learning Module-I (HSS) at T.Y. B.Tech. – I**

   Curriculum for Humanities and Social Sciences, Self Learning Module-I (HSS) is common for all under graduate engineering programs.

   A. Student can select & enroll a Self Learning Module-I (HSS) Course from P.A.H. Solapur University, Solapur Course List (SL31-A) and appear for university examination.

SL31-A: P. A. H. Solapur University, Solapur: HSS Course List

1. Economics	4. Stress and Coping
2. Intellectual Property Rights for Technology Development and Management	5. Professional Ethics & Human Value
3. Introduction to Sociology	

OR

B. Student can select and enroll for university approved minimum eight weeks NPTEL HSS course (SL31-B), complete its assignments and appear for certificate examination conducted by NPTEL. The list of courses as shown in Table SL31-B will be updated from time to time by University authorities. Latest updated list will be valid for selection of self learning Module-I (HSS) courses

More details about NPTEL are available at http://nptel.ac.in.

SL31-B: University approved NPTEL- HSS course List

1. Soft skills	15. Management of Inventory Systems
2. Introduction to Modern India Political Thought	16. Economic Growth and Development
3. Intellectual Property	17. Ethic in Engineering Practice
4. Technical English for Engineers	18. Corporate Social Responsibility
5. Developing Soft Skills and Personality	19. Marketing Management –I
6. Educational Leadership	20. Marketing Research and Analysis
7. Microeconomics: Theory & Applications	21. Selected Topics in Decision Modeling
8. Engineering Economics	22. Innovation, Business Models and Entrepreneurship
9. Human Resource Development	23. Simulation of Business Systems: An Applied Approach
10. Project Management for managers	24. Sustainability through Green Manufacturing Systems: An Applied Approach
11. Data Analysis and Decision Making - I	25. Total Quality Management - I
12. E-Business	26. Introduction to Operations Research
13. Working Capital Management	27. Knowledge Management
14. Industrial Safety Engineering	

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
FACULTY OF SCIENCE & TECHNOLOGY
Credit System Structure of Third Year B.Tech. (CSE) wef. 2020-2021
Semester – II

Course Code	***Theory Course Name***	***Hrs./Week***			***Credits***		***Examination Scheme***		
		L	***T***	***P***		***ISE***	***ESE***	***ICA***	***Total***
CS321	Compiler Construction	4	--	--	4	30	70	--	100
CS322	Unix Operating System	3	--	--	3	30	70	--	100
CS323	Computer Organization and Architecture	3	--	--	3	30	70	--	100
CS324	Artificial Intelligence	3	--	--	3	30	70	--	100
CS325	Mobile Application Development	2	--	--	2	25	--	--	25
CS326A to CS326C	Elective – I	3	--	--	3	30	70	--	100
SL32	Self Learning Module II (Technical)	--	--	--	2	--	50	--	50
	Sub Total	18	--	--	20	175	400	--	575
Course Code	***Laboratory Course Name***								
							ESE	***ICA***	
							POE	***OE***	
CS321	Compiler Construction	--	--	2	1		--	--	25	25
CS322	Unix Operating System	--	--	2	1		50	--	25	75
CS324	Artificial Intelligence	--	--	2	1		--	--	25	25
CS325	Mobile Application Development	--	--	2	1		50	--	25	75
CS326A to CS326C	Elective – I	--	--	2	1		--	--	25	25
CS327	Mini Project	--	--	2	1		--	50	25	75
	Sub Total	18	--	12	6		100	50	150	300
	Grand Total	**18**	**--**	**12**	**26**	**175**	**550**		**150**	**875**

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

Elective-I

CS326A - Object Oriented Modelling and Design
CS326B - $ Artificial Neural Network
CS326C - $ Data Science

$ - The theory courses for Computer Sci. and Engg. and Information Technology are same, therefore paper for ESE will be common to both.

Note :

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 vacation/s 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. 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.

4. Mini Project shall consist of developing software, based on various tools &technologies.

5. Project groups shall not be of more than **five** students.

6. **Self-Learning Module II at T.Y. B.Tech. – II (HSS)**

   A. Student can select a Self Learning Module II (Technical Course) from Course List (SL32) and appear for university examination.

SL32 : Self Learning Module-II (Technical)

SL32A - UI or UX technology
SL32B - Software Licensing and Practices

OR

B. Student can select & enroll for university approved minimum eight week technical course from various NPTEL technical courses, complete its assignments and appear for certificate examination conducted by NPTEL.

BOS Chairman / Coordinator will announce the list of approved NPTEL online courses of minimum eight weeks duration for 'Self Learning Module-II (Technical)' on commencement of the Sem-II of respective academic year from the available NPTEL courses through university system and will make available to student through University / institute website.

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B.Tech. (Computer Science & Engineering)
Semester – I
CS311 – SYSTEM PROGRAMMING

Teaching Scheme	**Examination Scheme**
Lectures – 3 Hrs./week, 3 Credits	ESE – 70 Marks
Practical – 2 Hrs./week, 1 Credit	ISE – 30 Marks
	ICA - 25 Marks

Introduction: This course introduces Language Processing activities, which helps to understand the basics of design and development of various professional languages, along with understanding of all the system software involved in executing a particular code written in a particular language.

Course Prerequisite: Students should have knowledge of Data Structures, Computer Organization, Microprocessors, Advanced C Concepts.

Course Outcomes:

Student will be able to

1. Describe the basic principles of system software and tools.
2. Implement Assembler and Macros to provide program generation facilities.
3. Use LPDT tools for a relevant problem to generate a scanner and parser.
4. Apply linkers and loaders for execution of a program.

SECTION - I

Unit 1 - Language Processors (8 hrs)

Introduction, language processing activities, Fundamentals of language processing, Fundamentals of language, Specification, language Processor development tools, UNIX programming Tools-lex & yacc, Recognizing words with Lex, Parser lexer communication, the parts of Speech lexer, A Yacc parser, The rules section of yacc, running lex and yacc on Unix.

Unit 2 - Assemblers (7 hrs)

Elements of assembly language programming, A simple assembly scheme, Pass structure of assemblers, design of a two pass assembler, A single pass assembler for IBM PC.

Unit 3 - Macros and Macro Processors (6 hrs)

Macro definition and call, Macro Expansion, Nested macro calls, Design of Macro preprocessor-Design overview.

SECTION – II

Unit 4 - Compilers and Interpreters (7 hrs)

Aspects of compilation, compilation of expressions, code optimization, Static and dynamic memory allocation, Memory allocation in block structured languages (Scope Rules, Memory allocation and access, Dynamic pointer), Interpreters

Unit 5 - Linkers **(6 hrs)**

Relocation and linking concepts, design of a linker, Self-relocating programs, linking for overlays.

Unit 6 - Loaders **(7 hrs)**

Function of loader, General loader scheme, Absolute loader, Relocating loader, Direct linking loader, Dynamic loading, Design of direct linking loader.

Internal Continuous Assessment (ICA) :

ICA consists of minimum 8 to 10 experiments based on the following guidelines.

1. Design Lex specifications for the tokens – keywords, identifiers, numbers, operators, white spaces.
2. Implementation of simple Lexical Analyzer in C which will generate the different tokens.
3. Implementation of syntax recognizer using grammar rules.
4. Simulation of text editor.
5. Introduction of TASM.
6. Symbol Table generation for *.c or *.asm file.
7. Design and Implementation of two pass assembler.
8. Design and Implementation of Single pass assembler.
9. Implementation of Macros.
10. Implementation of Nested macros.
11. Implementation of Toy-code generator.
12. Simulation of linkers.
13. Simulation of loaders.

Text Books:

1. System Programming and operating systems, D.M. Dhamdhere, 2nd Edition (TMGH) (Unit-1,2,3,4,5)
2. System Programming, J. J. Donovan, Mc-Graw Hill, (Unit 6)
3. Unix Programming Tools – lex & yacc , John R. Levine, Tony Mason & Doug Brown, (O"REILLY) (Unit 1)

Reference Books:

1. System Software - An Introduction to Systems Programming, Leland L. Beck, 3rd Edition (Pearson Education)
2. System Programming with C and Unix, Adam Hoover, Pearson,2010
3. Language Implementation Patterns, Terence Parr, SPD,2009

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B. Tech. (Computer Science & Engineering)
Semester – I
CS312 – OPERATING SYSTEMS

Teaching Scheme	**Examination Scheme**
Lectures– 3 Hrs./week, 3 Credits	ESE – 70 Marks
	ISE – 30 Marks

Introduction:

This course introduces Fundamentals and basic knowledge of an operating system. It also covers the details Process Management, deadlock, Memory Management and IO subsystems.

Course Prerequisite:

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

Course Outcomes:

Students will be able to:

1. Comprehend the features of operating system to formulate its role and responsibilities.
2. Analyze the principles of concurrency and synchronization to provide solution to the concurrent programs.
3. Simulate process scheduling and memory management techniques for CPU performance.

SECTION-I

Unit 1 - Introduction **(5 hrs)**

Operating system definition, Simple Batch System, Multiprogrammed Batch System, Time Sharing System, Personal Computer System, Parallel System, Real Time System, and System Calls.

Unit 2 - Process **(6 hrs)**

Process Concept, Process Scheduling, Operations on processes, Cooperating Processes, Threads, Inter-Process communication

Unit 3 - Process Scheduling **(6 hrs)**

Basic concept, Scheduling Criteria, Scheduling Algorithms, Multiple processor scheduling,

Unit 4 - Inter-process synchronization **(5 hrs)**

Background, The critical section problem, Peterson's algorithm, Synchronization Hardware, Semaphores, Classical problems of synchronization, Monitors.

SECTION-II

Unit 5 - Deadlocks **(7 hrs)**

System model, Deadlock characterization, Methods for handling deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from deadlock, combined approach to deadlock.

Unit 6 - Memory Management **(6 hrs)**

Background, Logical Versus Physical Address space, Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with paging.

Unit 7 - Virtual Memory **(5 hrs)**

Background, Demand paging, Page replacement, Page replacement algorithms, Allocation of frames, thrashing (Only concept).

Unit 8 - IO System **(4 hrs)**

Overview, I/O hardware, Application I/O interface, Mass Storage Structure – Disk Scheduling (FCFS Scheduling, SSTF Scheduling, SCAN Scheduling, C-SCAN Scheduling, LOOK Scheduling)

Text Books:

1. Operating System concepts, Silberschatz, Galvin, 7th or 8th Edition (John Wiley).
2. Operating Systems: Internals and Design Principles by William Stallings, 5th Edition (PHI).
3. The design of Unix Operating Systems- Maurice J. Bach(PHI)

Reference Books:

1. Operating system with case studies in UNIX, Netware and Windows NT by Achyut Godbole (TMGH).
2. Operating Systems, Deitel, Deitel, Choffnes, 3rd Edition, by Pearson Education.

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B.Tech. (Computer Science and Engineering)
Semester – I
CS313 - SOFTWARE ENGINEERING

Teaching Scheme	**Examination Scheme**
Lecture: 3 Hrs/week, 3 Credits	ESE: 70 Marks
	ISE: 30 Marks

Introduction : Current Software engineering methods and techniques have made us much better at building large and complex systems than we were. However, there are still too many projects that are late, over budget, and do not deliver the software that meets customer's needs. The main aim of introducing this course is to understand the methods, processes, techniques, and approaches which are required to develop high-quality software products within schedule and budget. On top of that this course ensures understanding of the complete Software Development Life Cycle (SDLC) for the development of software products as per the customer's needs. Further, it ensures the knowledge of various quality standards used in the software system and the Agile Project Management Process.

Course Prerequisite:

Student shall have undergone a course on Object Oriented Programming through C++. An understanding of Object Oriented Analysis and Design and Programming skills.

Course Outcomes:

At the end of the course Student will be able to

1. Select and apply the appropriate lifecycle model for software development.
2. Prepare SRS and SDS accordingly for a given problem.
3. Select and apply appropriate software testing method.
4. Ensure the quality of a product by applying the quality management process.

SECTION-I

Unit 1 - Introduction to Software Engineering (11 Hrs)

Introduction, The Problem Domain, Software Engineering Challenges and Approach, Software Process, Characteristics of Software Process, Software Development Process Models: Waterfall model, Prototype model, Iterative development model: Incremental Model, Spiral model, Rational unified Process model, Time Boxing model, Agile process model.

Unit 2 - Software Requirement Analysis & Specification (6 Hrs)

Need of SRS, Characteristics of Good SRS, Requirement Process, Requirements specification, Functional Specification with Use Cases, Other Approaches for Analysis: Data Flow Diagram, Entity Relationship Diagram,

Unit 3 - Software Architecture and Design (8 Hrs)

Introduction to Software Design, Software Architecture: Role of Software Architecture, Architecture Views, Component & Connector View, Architecture Style for Component & Connector view, Documenting Architecture Design, Design Concepts: Design Principles, Conceptual Design and

Technical Design, Coupling, Cohesion, Open Closed Principle, Function-Oriented Design, Object Oriented Design, High Level Design, Detailed Design, Verification, Metrics.

SECTION-II

Unit 4 - Testing **(4 Hrs)**

Testing Fundamentals, Testing Process, Black-Box Testing, White-Box Testing, Object-Oriented Software testing methods, Functional testing, Unit testing, System testing, User satisfaction testing.

Unit 5 - Project Planning and Management **(8 Hrs)**

Project management process, The Inspection and Audit Process, Software Configuration Management process, Effort estimation, Project Schedule and Staffing, Quality planning: Quality Concepts, Qualitative quality management planning. CMM project management process, Risk Management Planning, Project Monitoring Plan, Detailed Scheduling.

Unit 6 - Agile Project Management **(8 Hrs)**

Introduction to APM, Implementation, Iterative Project Management Life Cycle, Adaptive Project Management Life Cycle, Adaptive & Integrating the APM toolkit, The Science of Scrum, New Management Responsibilities.

Text Books:

1. An Integrated Approach to Software Engineering, Pankaj Jalote, 3rd Edition (Narosa Publishers)
2. Effective Project Management Traditional, Agile, Extreme, Robert K. Wysocki, 6th Edition, WILEY INDIA
3. Software project management in practice,Pankaj Jalote Pearson India Ltd

---

Reference Books :

1. Software Engineering, Ian Sommerville, 6th edition, Pearson education Asia
2. Software Engineering Fundamentals, Ali Behforooz and Frederick j. Hudson (Oxford University Press).
3. Project Management with Scrum, Ken Schwaber.
4. Software Engineering-A precise approach, Pankaj Jalote Wiley Precise Precise Textbook

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Third Year B.Tech (Computer Science & Engineering)
Semester – I
CS314 - DATABASE ENGINEERING

Teaching Scheme	**Examination Scheme**
Lectures: 4 Hrs/week, 4 Credits	ESE: 70 Marks
Practical: 2 Hrs/week, 1 Credit	ISE: 30 Marks
	ICA: 25 Marks
	POE: 50 Marks

Introduction:

In today's data-driven economy, no computer science or business curriculum would be complete without a course in databases and data management system. This course is designed to introduce graduate students to the foundations of database systems, focusing on basics such as the relational algebra, data model, and Normalization.

It also provides students with theoretical knowledge and practical skills in the use of databases and database management systems with the help of Structured Query language (SQL).It ends with covering database transaction and recovery concepts. Upon completion, students should be able to design and implement normalized database structures by creating simple database.

Course Perquisites:

Math foundations: elementary set theory, concepts of relations and functions

Data structures: trees, B-trees, linear data structures, dictionaries, graphs.

Algorithms: Basic algorithm design methods and techniques for algorithm complexity analysis

---

Course Outcomes:

At the end of the course students will be able to,

1. Demonstrate basics of database systems and design database using Entity-relationship model for real time application.
2. Design database using relational model for real time application and Formulate SQL.
3. Analyze a database design & apply normalization.
4. Create indices for faster retrieval.
5. Apply transaction management for maintaining database consistency.

SECTION–I

Unit 1: Introduction (5 Hrs)

Database-System Applications, Purpose of Database Systems, View of Data, Database Languages, Relational Databases, Database Design, Data Storage and Querying, Transaction Management, Database Architecture, Data Mining and Information Retrieval, Specialty Databases, Database Users and Administrators, History of Database Systems.

Unit 2: Database Design and the E-R Model (7 Hrs)

Overview of Design Process, The Entity-Relationship Model, Constraints, Removing Redundant Attributes In Entity Sets, E-R Diagrams, Reduction to Relational Schemas, E-R Design Issues, Extended E-R Features.

Unit 3: Relational Model **(8 Hrs)**

Structure of Relational Databases, Database Schema, Keys, Schema Diagrams, Relational Query Languages, Relational Algebra, Tuple Relational Calculus, Domain Relational Calculus, Structured Query language (SQL)-Overview, SQL Data Definition, Basic Structure of SQL Queries, Additional Basic Operations, Set Operations, Null Values, Aggregate Functions, Nested Sub queries, Modification of the Database, Join Expressions, Views, Transactions, Integrity Constraints, SQL Data Types and Schemas, Authorization.

Unit 4: Relational Database design **(7 Hrs)**

Features of Good Relational Designs, Atomic Domains and First Normal Form, Decomposition using Functional Dependencies, Functional Dependency Theory, Algorithms for Decomposition, Decomposition using Multi-valued Dependencies.

SECTION – II

Unit 5: Indexing and Hashing **(6 Hrs)**

Basic Concepts, Ordered Indices, B+-Tree Index Files, B+-Tree Extensions, B Tree Index Files, Multiple Key Access, Static Hashing, Dynamic Hashing, Comparison of Ordered Indexing and Hashing, Bitmap Indices, Index Definition in SQL.

Unit 6: Transactions **(6 Hrs)**

Transaction Concept, A Simple Transaction Model, Storage Structure, Transaction Atomicity and Durability, Transaction Isolation, Serializability, Transaction Isolation and Atomicity, Transaction Isolation Levels, Implementation of Isolation Levels, Transactions as SQL Statements.

Unit 7: Concurrency Control **(6 Hrs)**

Lock-Based Protocols, Deadlock Handling, Multiple Granularity, Timestamp-Based Protocols, Validation-Based Protocols.

Unit 8: Recovery System **(8 Hrs)**

Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithm, Buffer Management, Log-Based Recovery, Shadow Paging, Recovery with Concurrent Transactions, Buffer Management, Failure with Loss of Nonvolatile Storage.

Internal Continuous Assessment (ICA) :

ICA shall consist of following laboratory assignments:

- For any real life application:
  - Draw an E-R diagram and create a data dictionary for the same.
  - Basic SQL DDL and DML commands
  - Nested sub queries, Joins and Set operations
  - Views, Integrity constraints and Authorization
  - Identify set of functional dependencies, find canonical cover and closure of functional dependency
  - Convert the created database into 1NF, 2NF, 3NF and BCNF
  - Program to create B+ tree index

- Program to implement dynamic hashing on the database

- Program to simulate log based protocol using immediate or deferred database modification.
- Program to simulate any one concurrency control protocol.

Text Books:

1. Database system concepts by Abraham Silberschatz, Henry F. Korth, S. Sudarshan (6^{th} edition, McGraw Hill International Edition).
2. Database system concepts by Peter Rob, Carlos Coronel (Cengage Learning).
3. Fundamentals of Database systems by Ramez El Masri, S. B. Navathe (Pearson Education)

Reference books:

1. Database Management Systems by RamkrishnanGehreke (Tata McGraw Hill).
2. Principles of Database Systems by J. D. Ullman (Galgotia Publications)
3. Advanced Database Management System by Rini Chakrabarti, Shilbhadra Dasgupta (Dreamtech Press Publication).
4. SQL The Complete Reference(Third Edition, McGraw Hill International Edition)

Modalities for conducting End Semester Examination/Practical Oral Examination (POE):

- Practical and Oral Examination will be conducted by a panel of examiners assigned by university. A pair of examiners shall assess a batch @36 students in a day.
- The chairman shall prepare problem statements for a batch adhering to following guidelines:
  1. At least Four Problem statements shall be set for a batch.
  2. Problem statements shall be set in the context of course outcomes as defined in the course.
  3. Problem Statements shall not be direct statements stating implement concept/topic etc.
  4. Problem Statements shall be based on real world problem/use case/scenario etc.
  5. Problem statement shall be at the minimum cognitive level of 'Apply' & above.
  6. Problem Statements must be well described with no ambiguity and shall be of unseen nature.
  7. Problem statements formulated shall be solvable by faculty in 2 hours and average students in 2.5 hrs in the examination duration.

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B.Tech. (Computer Science and Engineering)
Semester – I
CS315 - DESIGN AND ANALYSIS OF ALGORITHM

Teaching Scheme	**Examination Scheme**
Lectures– 3 Hrs./week, 3 Credits	ESE –70 Marks
Practical– 2 Hrs./week, 1 Credit	ISE – 30 Marks
	ICA –25 Marks

Introduction :

This course introduces the algorithms, strategies of algorithm and analysis of algorithm which will help to compare and determine good algorithm.

Course Prerequisite:

Student should have knowledge of basic programming. They should also have basic knowledge of data structure and graph theory.

Course Outcomes:

At the end of the course, students will be able to -

1. Derive time and space complexity of a given algorithm
2. Select appropriate algorithm design paradigm for a problem.
3. Apply algorithm design paradigm for a problem.
4. Describe and distinguish complexity classes of problems

SECTION-I

Unit 1 - Introduction (8 Hrs.)

Algorithm Specification: Pseudo code Conventions, Recursive Algorithm, Performance Analysis: Space Complexity, Time Complexity, Calculating worst case, best case and average case complexities, complexities Asymptotic Notations, Performance Measurement

Unit 2 - Divide and Conquer (7 Hrs.)

The general method, Binary search, Finding the maximum and Minimum, Quicksort, Selection Sort, Merge Sort.

Unit 3 - The Greedy method (8 Hrs.)

The general method, Knapsack Problem, Job Sequencing with deadlines, Minimum –cost spanning trees – Prim's and Kruskal's Algorithms, Optimal storage on tapes, Optimal merge patterns, Single source shortest paths

SECTION-II

Unit 4 - Dynamic Programming **(8 Hrs.)**

The general method, Multistage graphs, All pair shortest paths, Optimal binary search trees, 0/1 Knap sack, Reliability design, The Traveling Sales person problem. Flow shop scheduling

Unit 5 - Backtracking **(7 Hrs.)**

The general method, 8-queen problem, sum of subsets, Knapsack Problem, Hamilton Cycle, and Graph Coloring.

Unit 6 - NP-Hard and NP-Complete problems **(7 Hrs.)**

Tractable and Intractable Problems: Computability. The Halting problem, Computability classes – P, Np- class, NP-complete and NP-hard, Standard NP-complete problems, NP-Hard Problem (Only Basics problems).

Internal Continuous Assessment (ICA) :

ICA shall consist of minimum ten practical assignment problems.

1) The nature of the problems shall be with objectives to assess student's ability to
    a. Compare and choose an appropriate algorithm design paradigm on time and space complexity
    b. Apply algorithm design paradigm to provide a solution to the problem using either C, C++, Python, Java or any other programming language the student is proficient in.
    c. Effectively assess performance of provided solutions w.r.t programming language's runtime implementation.

Text Book:

1. Fundamentals of Computer Algorithms, Horowitz, Sahni & Rajasekaran (Galgotia Publications)
2. Fundamental of Algorithm, Gilles Brassard, Paul Bratley (Pearson Publication)
3. Introduction to Algorithms, Thomas Cormen (Pearson Publication)

Reference books :

1. Introduction to Design and Analysis of Algorithm, Goodman (McGrawhill)
2. Design and analysis of algorithms, Aho, Hopfcraft and Ullman (Addison wesley)
3. Design & Analysis of Algorithms, Sharma, Khanna Publishing House, N.Delhi
4. Design& Analysis of Algorithms, S. Sridhar, Oxford

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B.Tech. (Computer Science & Engineering)
Semester – I
CS316 - PYTHON PROGRAMMING

Teaching Scheme	**Examination Scheme**
Lectures: 2 Hrs./Week, 2 Credits	ISE : 25 Marks
Practical : 2 Hrs./Week,1 Credit	ICA : 25 Marks
	POE : 50 Marks

Introduction :

Python is a popular, general-purpose, multi-paradigm, open-source, scripting language. It is designed to emphasize code readability – has a clean syntax with high level data types. It is suited for interactive work and quick prototyping, while being powerful enough to write large applications. This course introduces basic Python language syntax and Python standard Library.

Course Prerequisite:

Student should have knowledge of basic programming.

Course Outcomes :

At the end of this course, students will be able to

1. Write Python scripts using procedure and object oriented approach of writing a computer program.
2. Exhibit ability to use Python's standard library packages to provide solution to a given problem.
3. Test and debug python script for a given problem.

Unit 1 - Introduction to Python (2 hrs)

Introducing the Python Interpreter, Program Execution, Execution Model Variations, The Interactive Prompt, System Command Lines and Files. Syntactic and semantic differences between Python 2.x and Python3.x.

Unit 2 - Introduction to Python Programming Constructs (4 hrs)

Data types and variables, Collection data types, Control structures, loops and functions, Lambdas, Generators, Exception Handling, String handling, Scope of variables, Modules, Packages, Command line arguments. Built-in: Functions, Constants, Types, Exceptions.

Unit 3 - Introduction to Object Oriented Programming in Python (4 hrs)

Classes, Instance Objects, Method Objects, Class and Instance Variables, Attributes and methods, Inheritance and polymorphism

Unit 4 - Python Standard Library Modules and Packages -1 (6 hrs)

Common string operations, Regular expression operations, Basic date and time types, General calendar-related functions, Container datatypes, Efficient arrays of numeric values, Dynamic type creation and names for built-in types, Shallow and deep copy operations, Mathematical functions, Generate pseudo-random numbers, Functional Programming Modules, File and Directory Access

Unit 5 - Python Standard Library Modules and Packages -2 (6 hrs)

Data Persistence: Python object serialization, DB-API 2.0 interface for SQLite databases. Work with ZIP archives, CSV File Reading and Writing, Configuration file parser, Logging facility for Python.

Concurrent Execution: Thread-based parallelism, Process-based parallelism, Context Variables, Asynchronous I/O.

Low-level networking interface, JSON encoder and decoder, URL handling modules, urlib, HTTP modules, HTTP protocol client,

Unit 6 - Testing, Debugging and Profiling: (4 hrs)

Testing output, Unit tests in Python, Handling Multiple exceptions, Creating custom exceptions, Debugging programs, Unit testing, Measure execution time of small code snippets, Creation of virtual environments, System-specific parameters and functions and profiling Python scripts.

ISE Evaluation for the course will consists of three programming (hands on) tests.

Internal Continuous Assessment (ICA): Minimum 12 assignments based on above topics.

- The assignments should test and develop student's practical proficiency and ability to use Python standard library modules and packages efficiently in writing effective code for varied applications scenarios & requirements, use cases.
- Use of IDEs like PyCharm, Eclipse with PyDev, Jupyter Notebook for Interactive development and debugging of Python applications is highly recommend to enhance hands on skills in Python Programming of Students.
- Every assignment shall be performed under Python 2.x or 3.x runtime environment configured using any of the following tools 1) pyenv 2) virtualenv 3)Anaconda

---

Text Book:

1. Programming in Python 3, Mark Summerfield, Second Edition
2. Python Programming, Reema Thareja, Pearson
3. Introduction to Computing and Problem Solving with Python, J. Jose, Khanna Publications
4. Learning Python, MarkLutz, 5th edition

Reference Books:

1. Python Cookbook, David Beazley and Brian K. Jones, Third Edition, Shroff Publishers & Distributors Pvt. Ltd., ISBN :978-93-5110-140-6
2. Testing Python, David Sale, Wiley India (P) Ltd., ISBN :978-81-265-5277-1
3. Programming Python (English), MarkLutz, 4th Edition
4. Taming Python by Programming, Jeeva Jose, Khanna Publishing House

---

e-resources :

1. Python 2.7.16 documentation - https://docs.python.org/2/
2. Python 3.7.3 documentation - https://docs.python.org/3/

Modalities for conducting End Semester Examination/Practical Oral Examination (POE):

- Practical and Oral Examination will be conducted by a panel of examiners assigned by university. A pair of examiners shall assess a batch @36 students in a day.
- The chairman shall prepare problem statements for a batch adhering to following guidelines:
  1. Minimum four Problem statements shall be set for a batch.
  2. Problem statements shall be set in the context of course outcomes as defined in the course.
  3. Problem Statements shall not be direct statements stating implement concept/topic etc.
  4. Problem Statements shall be based on real world problem/use case/scenario etc.
  5. Problem statement shall be at the minimum cognitive level of 'Apply' & above.
  6. Problem Statements must be well described with no ambiguity and shall be of unseen nature.
  7. Problem statements formulated shall be solvable by faculty in 2 hours and average students in 2.5 hrs in the examination duration.

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B.Tech. (Computer Science & Engineering)
Semester – I
CS317 - JAVA PROGRAMMING

Teaching Scheme	**Examination Scheme**
Lectures: 2 Hrs/Week, 2 Credits	ISE : 25 Marks
Practical : 2 Hrs/Week,1 Credit	ICA : 25 Marks
	POE : 50 Marks

Introduction :

The course introduces Java language's syntax and object-oriented programming paradigms from the perspective of Java language. Further, the course thoroughly touches upon the vital aspects of the usage of Java runtime library packages' classes and methods.

Course Prerequisite:

Students must be familiar with basic programming languages like C.

Course Outcomes :

At the end of this course students will be able to

1. Implement Object Oriented Programming paradigm using Java language.
2. Exhibit the ability to use Java runtime library APIs to provide a solution to a given problem.
3. Test and debug a Java program for a given problem.

Unit 1 - Basics of Java and Strings in Java (2 hrs)

Basics: Java Runtime Environment (Oracle JDK, OpenJDK), Naming Conventions and Java profilers.

Basics: Variables, Operators, Expressions, Statements, Blocks, Control flow Statements, Input and Output, Data Types, Arrays, Type Casting.

Fundamentals: String Class and Methods, Immutability of Strings, String Buffer Class and Methods, String Builder class and Methods.

Unit 2 - Introduction to OOPs (4 hrs)

Objects and Classes, Fields and Methods, Abstraction, Encapsulation, Inheritance, Polymorphism, Type Compatibility and Conversion, Overriding Methods, Access control, Modifiers, Constructors, Abstract classes, Nested classes, Packages, Wrapper classes, Interfaces, Object Life time & Garbage Collection.

Unit 3 - Exceptions, Error Handling and Basic IO (6 hrs)

Exceptions and Error Handling: Exceptions and Errors, Catching and Handling Exceptions, The try Block, The catch Blocks, The finally Block, Throwing Exceptions, Chained Exceptions , Custom Exceptions. JUnit Testing Framework.

Basic I/O: I/O Streams, Byte Streams, Character Streams, Buffered Streams, Scanning and Formatting, Data Streams, Object Streams , File I/O Classes: Reading, Writing, and Creating Files and Directories.

Unit 4 - Java Collections Framework **(6 hrs)**

Introduction, The Arrays Class, Searching and sorting arrays of primitive data types, Sorting Arrays of Objects, The Comparable and Comparator Interfaces, Sorting using Comparable & Comparator, Collections: Lists, Sets, Maps, Trees, Iterators and Collections, The Collection Class.

Unit 5 - Multithreading and Networking **(6 hrs)**

Multithreading: Creating Threads, Thread scheduling and priority, Thread interruptions and synchronization.

Network Programming: InetAddress, URLs, Socket (TCP & UDP) communication in Java, Servlet Programming

Unit 6 - GUI Programming using Swing: Swing package, Layouts, Events, Listeners and Event handling, and Swing Components. **(3 hrs)**

Unit 7 - JDBC: Introduction to JDBC, JDBC Drivers & Architecture, CRUD operations Using JDBC API. **(3 hrs)**

ISE Evaluation: ISE Evaluation for the course will consists of three programming (hands on) tests.

Internal Continuous Assessment (ICA):

ICA shall consist of minimum 15 practical assignment problems.

The assignments should test and develop student's practical proficiency and ability to use Java API Classes correctly for writing code for varied applications scenarios & use case requirements.

Use of IDEs like BlueJ, Eclipse, Netbeans or any other FOSS alternative for Interactive development and debugging of Java applications is highly recommend to enhance hands on skills in Java Programming of Students.

Text Books:

1. Head First Java, Kathy Sierra, Bert Bates, O'Reilly Publication
2. The JavaTM Programming Language, Ken Arnold, James Gosling, David Holmes, Pearson Publication
3. Core Java for Beginners, Rashmi Kanta Das, Vikas Publishing House Pvt Ltd.
4. Programming with Java, Balaguruswamy, TMH
5. Internet and Java Programming, Tanweer Alam, Khanna Publishing House

---

Reference Books:

1. The Java Language Specification, Java SE 8 Edition Book by James Gosling, Oracle Inc.
2. Java: The Complete Reference 8 Edition - Herbert Schildt , Tata McGraw - Hill Education
3. Head First Servlets and JSP – Bryan Bosham, Kathy Sierra, Bert Bates, O'Reilly Publication
4. The JavaTM Tutorials. Oracle Inc.
5. Java Server Programming for Professionals - Ivan Bayross, Sharanam Shah, Cynthia Bayross
6. and Vaishali Shah, Shroff Publishers and Distributors Pvt. Ltd, 2nd Edition

---

e-resources :

1. http://docs.oracle.com/javase/specs/
2. http://docs.oracle.com/javase/tutorial/

Modalities for conducting End Semester Examination/Practical Oral Examination (POE):

- Practical and Oral Examination will be conducted by a panel of examiners assigned by university. A pair of examiners shall assess a batch @36 students in a day.
- The chairman shall prepare problem statements for a batch adhering to following guidelines:
  1. At least Four Problem statements shall be set for a batch.
  2. Problem statements shall be set in the context of course outcomes as defined in the course.
  3. Problem Statements shall not be direct statements stating implement concept/topic etc.
  4. Problem Statements shall be based on real world problem/use case/scenario etc.
  5. Problem statement shall be at the minimum cognitive level of 'Apply' & above.
  6. Problem Statements must be well described with no ambiguity and shall be of unseen nature.
  7. Problem statements formulated shall be solvable by faculty in 2 hours and average students in 2.5 hrs in the examination duration.

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B.Tech. (Computer Science & Engineering)
Semester-I
SELF LEARNING MODULE – I (H.S.S.)
SL31A-1 ECONOMICS

Teaching Scheme	**Examination Scheme**
Credits :- 2 Credits	**ESE:** 50 Marks

Course Outcomes:

Upon completion of this course, students will be able to

1. Identify the Basic Economic problems, Resource Constraints.
2. Apply various theories of economics for economic growth.
3. Identify causes of Inflation consequence and remedies.
4. To assess the impact of International Trade, foreign exchange on Indian economy.

Unit 1: Introduction

History of Economic thought, Basic Economic problems, Resource Constraints and Welfare maximization

Nature of Economics: Positive and Normative Economics, Micro and Macro Economics, Basic concepts in Economics, The role of State in economic activity, Market and Government failures, New economic Policy in India.

Unit 2: Theories of Economics

Theory of utility and consumer's choice, Theories of Demand, supply and market equilibrium, Theories of firm, production and costs, Market structures, Perfect and imperfect competitions, oligopoly, monopoly.

Unit 3: Macroeconomics

An overview of Macroeconomics, measurement and determination of national income, Consumption, saving and investment.

Unit 4: Banking & Inflation.

Commercial and Central Banking, Relationship between money, output and prices. Inflation causes, consequences and remedies.

Unit 5: International Influences on Economics

International Trade, foreign exchange and balance payments, stabilization policies, Monetary, Fiscal and exchange rate policies.

ASSIGNMENTS

Students shall complete five assignments, based on syllabus. (One assignment for every unit of the syllabus)

In addition to the above, the institute may prescribe additional modes of assessment such as Unit test, Quiz, Presentation, Course seminar etc. for ensuring continuous assessment of the students.

TEXT BOOKS

1. Economics: P.A. Samuelson & W.D Nordhaus (McGraw Hill, New York, 1995.)
2. Modern Microeconomics : A. Koutsoyiannis (Macmillan,1975)

REFERENCE BOOKS

1. Microeconomics: R. Pindyck and D.L. Rubinfield. (Macmillan New York, 1989
2. Microeconomics: Gordon, 4th edition, Little Brown & Co., Boston,1987.
3. The Organization of Industry: William F. Shughart II, Richard D. Irwin, Illinois, 1990.

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B.Tech. (Computer Science & Engineering)
Semester-I
Self Learning Module – I (H.S.S.)

SL31A-2 INTELLECTUAL PROPERTY RIGHTS FOR TECHNOLOGY DEVELOPMENT AND MANAGEMENT

Teaching Scheme	**Examination Scheme**
Credits :- 2 Credits	**ESE:** 50 Marks

Course Outcomes:

Upon completion of this course, students will be able to,

1. Appreciate the intellectual property rights coming out of research and intellectual works
2. Demonstrate their knowledge about the process of acquiring the patents and copyrights for the innovative works.
3. Elaborate the role of Indian IPR system and role of WTO in protecting Intellectual Property Rights
4. Avoid the plagiarism in their thesis, research papers etc. which can be questioned legally.

Unit 1:

Dynamics of Knowledge evolution, creation of ownership domains in the knowledge space using various instruments of IPR

Unit 2:

Outlines concepts of confidentiality and information security, explores their role in technology development and transfer integrating Intellectual Property in project planning, execution & commercialization,

Unit 3:

Discussion on the shifting paradigms of R&D and their linkage to IPR, Introduction to concepts of Valuation of IP & Value Realization,

Unit 4:

Comparison the Indian IPR system with international IPR frameworks especially in the context of WTO, followed by a few sessions on IPR litigations both for the enforcement of rights and business strategy.

Unit 5:

Discussion on contentious issues of current interest such as Biotechnology and Intellectual Property, Protection of Traditional Knowledge, IPR and Electronic Commerce, TRIPS and Access to Medicines, Copyright issues in creative works, etc.

ASSIGNMENTS

Students shall complete five assignments, based on syllabus. (One assignment for every unit of the syllabus)

In addition to the above, the institute may prescribe additional modes of assessment such as Unit test, Quiz, Presentation, Course seminar etc. for ensuring continuous assessment of the students.

TEXT BOOKS

1. Prabuddha Ganguli: Intellectual Property Rights Unleashing the Knowledge Economy. Tata McGraw Hill, New Delhi, 2001.
2. Prabuddha Ganguli: Gearing Up for Patents The Indian Scenario. Universities Press India Ltd., Hyderabad, 1998.
3. P. Narayan: Patent Law. Eastern Law Co., Calcutta.

REFERENCE BOOKS

1. Global Dimensions of Intellectual Property Rights in Science and Technology, Author: National Research Council , National Academies Press, 1993.
2. Technology Transfer: Intellectual Property Rights, C Sri Krishna, ICFAI University press (2008)

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B.Tech. (Computer Science & Engineering)
Semester-I
Self Learning Module – I (H.S.S.)
SL31A -3 INTRODUCTION TO SOCIOLOGY

Teaching Scheme	**Examination Scheme**
Credits :- 2 Credits	**ESE:** 50 Marks

Course Outcomes:

Upon completion of this course, students will be able to,

1. Interpret the effect of various social phenomena on sociology
2. Elaborate the role of urbanization on the society
3. Appreciate the need of social institutions for better society.
4. Assess the role of modernization, industrialization, environmental/ecological changes in the development of society.

Unit 1:

What is sociology, some sociological concepts: social structure, status, role, norms, values etc., Socialization, and culture and change.

Social stratification - various approaches and concept of social mobility.

Unit 2:

Population and society - Trends of demographic change in India and the world, Human Ecology, Trends of Urbanization in the developing countries and the world.

Unit 3:

Major social institutions - Family and marriage, caste and tribe and organizations:

i. Formal organization (bureaucracy)
ii. Informal Organization

Unit 4:

Processes of social change- Modernization (including Sanskritization), industrialization, environmental/ecological changes and development.

Unit 5:

Social movements - protest movements, reformist movement and radical movements in India.

ASSIGNMENTS

Students shall complete five assignments, based on syllabus. (One assignment for every unit of the syllabus)

In addition to the above, the institute may prescribe additional modes of assessment such as Unit test, Quiz, Presentation, Course seminar etc. for ensuring continuous assessment of the students.

TEXT BOOKS

1. Sociology, L. Broom, P. Selznick and D. Dorrock, 11th Edn. 1990 (Harper International).
2. Sociology: Themes and Perspectives, M. Haralambos, Oxford University Press, 1980.
3. General Introduction to Sociology, Guy Rocher, A. , MacMillan, 1982.

REFERENCE BOOKS

1. Social movements in India, vols. 1-2, 1984, M.S.A. Rao, Manohar Publications.
2. Society in India, David Mandelbaum, 1990, Popular Publications.
3. Social change in modern India, M.N. Srinivas, 1991, Orient Longman Publications.

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B.Tech. (Computer Science & Engineering)
Semester-I
Self Learning Module – I (H.S.S.)
SL31A -4 STRESS AND COPING

Teaching Scheme	**Examination Scheme**
Credits :- 1 Credit	**ESE:** 50 Marks

Course Outcomes:

Upon completion of this course, students will be able to,

1. Identify various sources and nature of a stress.
2. Elaborate the effects of medical, psychological and behavioral stress.
3. Appreciate the social support to mitigate the stress.
4. Adopt various stress management techniques.

Unit 1:

Concept of stress-current and historical status. The nature of the stress response.

Unit 2:

Common sources of stress biological, personality and environmental.

Unit 3:

Coping styles defensive behaviors and problem-solving. Consequences of stress - medical, psychological and behavioral.

Unit 4:

The role of social support in mitigating stress.

Unit 5:

Stress management techniques-relaxation, meditation, cognitive restructuring, self-control, bio-feedback and time management, Preparing stress profile of a student.

ASSIGNMENTS

Students shall complete five assignments, based on syllabus. (One assignment for every unit of the syllabus)

In addition to the above, the institute may prescribe additional modes of assessment such as Unit test, Quiz, Presentation, Course seminar etc. for ensuring continuous assessment of the students.

TEXT BOOKS

1. Walt, S. "Stress Management for Wellness". Harcourt Brace & Jovanovich, N.York, 1994.
2. D. Girdano and G. Everly., "Controlling Stress and Tension", Prentice-Hall, 1986.
3. Monat and R. Lazarus, "Stress and Coping: An Anthology", Columbia Univ. Press, 1985.

REFERENCE BOOKS

1. Weisman, "The Coping Capacity", Human Services Press, 1984.
2. Stress and Coping: The Indian Experience, D.M. Pestonjee, SAGE India; Second edition (1998)

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B.Tech. (Computer Science & Engineering)
Semester-I
Self Learning Module – I (H.S.S.)
SL31A - 5 PROFESSIONAL ETHICS & HUMAN VALUES

Teaching Scheme	**Examination Scheme**
Credits :- 2 Credits	**ESE:** 50 Marks

Course Outcomes:

Upon completion of this course, students will be able to,

1. Inculcate the human values in their behavior.
2. Demonstrate the Engineering ethics in their professional practice.
3. Practice the safety and responsibility and professional rights in their professional practice.
4. Incorporate the code of ethics of Global organizations such as ASME, ASCE, and IEEE

Unit 1: Human Values

Morals, Values and Ethics, Integrity, Work Ethics, Service Learning, Civic Virtue, Respect for others, Living Peacefully, Caring, sharing, Honesty, Courage, Valuing Time, Cooperation, Commitment, Empathy, Self-Confidence, Character, spirituality

Unit 2: Engineering Ethics

Senses of engineering ethics, Variety of Moral Issues, Types of inquiry, Moral Dilemmas Moral Autonomy, Kohlberg's Theory, Gilligan's Theory, Consensus and Controversy, Models of Professional Roles, Theories about Right Action, Self Interest , Customs and Religion.

Unit 3: Safety, Responsibilities and Rights

Safety and Risk, Assessment of safety and Risk, Risk Benefit Analysis and Reducing Risk, The Three Mile Island and Chernobyl Case Studies.

Collegiality and Loyalty, Respect for Authority, Collective Bargaining, Confidentiality, Conflicts of Interest, Occupational Crime, Whistle Blowing, Professional Rights – Employee Rights, Intellectual Property Rights (IPR) – Discrimination

Unit 4: Global Issues

Multinational Corporations, Environmental Ethics, Computer Ethics, Weapons Development, Engineers as Managers, Consulting Engineers, Engineers as Expert Witnesses and Advisors, Sample Code of Ethics of ASME, ASCE, IEEE, Institution of Engineers (India), etc.

ASSIGNMENTS

Students shall complete five assignments, based on syllabus. (One assignment for every unit of the syllabus)

In addition to the above, the institute may prescribe additional modes of assessment such as Unit test, Quiz, Presentation, Course seminar etc. for ensuring continuous assessment of the students.

TEXT BOOKS

1. Bayles, M.D.: Professional Ethics, California: Wadsworth Publishing Company, 1981.
2. Koehn, D.: The Ground of Professional Ethics, Routledge, 1995.
3. R.S. Naagarazan, A Text Book of Professional Ethics & Human Values, New Age International, 2006

REFERENCE BOOKS

1. Camenisch, P.F.: Grounding Professional Ethics in a Pluralistic Society, N.Y.: Haven Publications, 1983.
2. Wuest, D.E.: Professional Ethics and Social Responsibility, Rowman & Littlefield, 1994.

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B.Tech. (Computer Science & Engineering)
Semester – II
CS321 - COMPILER CONSTRUCTION

Teaching Scheme	**Examination Scheme**
Lectures: 4 Hrs/week, 4 Credits	ESE: 70 Marks
Practical : 2 Hrs/week, 1 Credit	ISE: 30 Marks
	ICA : 25 Marks

Introduction:

A compiler translates a program written in a high-level programming language that is suitable for human programmers into the low-level machine language that is required by computers.

Since writing a compiler is a nontrivial task, it is a good idea to split the compilation into several phases with well-defined interfaces. Conceptually, these phases operate in sequence, each phase except first phase taking the output from the previous phase as its input. Each phase is handled by a separate module.

This course provides an in-depth view of translation and optimization process. All phases required for translating a high-level language to machine language is covered in this course including scanning, parsing, intermediate-code generation, machine-code generation, register allocation and code optimization.

Course Prerequisite:

1. Theory of Computation
2. System Programming
3. Programming Language knowledge

Course Outcomes:

At the end of the course, students will be able to

1. Describe language translation and compiler design constructs.
2. Design and develop lexical analyzer and parser.
3. Apply optimization principles for generating code
4. Describe storage allocation strategies for memory allocation

---

SECTION-I

Unit 1 - Introduction to Compiling **(3 Hrs)**

Introduction, Compilers, Phases of a compiler, Compiler construction tools

Unit 2 - Lexical Analysis **(7 Hrs)**

Role of a Lexical analyzer, Input buffering, Specification and recognition of tokens, Finite automata implications, Designing a lexical analyzer generator

Unit 3 - Syntax Analysis **(10 Hrs)**

Role of Parser, Writing grammars for context free environments, Top-down parsing, Recursive descent and predictive parsers (LL), Bottom-Up parsing, Operator precedence parsing, LR parsers, SLR parsers, LALR parsers.

Unit 4 - Syntax Directed Translation **(8 Hrs)**

Syntax directed definitions, construction of syntax tree, Bottom-up evaluation of S-attributed definitions, L-attributed definitions, Top-down translation of inherited attributes, Bottom-up evaluation of inherited attributes, Analysis of syntax directed definitions.

SECTION-II

Unit 5 - Run Time Environments **(5 Hrs)**

Source language issues, storage organization and allocation strategies, Parameter passing, Symbol table organizations and generations, Dynamic storage allocations

Unit 6 - Intermediate Code Generation **(7 Hrs)**

Intermediate languages, declarations, Assignment statements, Boolean expressions, case statements Back patching, procedure calls, Back patching, procedure calls

Unit 7 - Code Generation **(8 Hrs)**

Issues in design of a code generator and target machine, Run time storage management, Basic blocks and flow graphs, Next use information and simple code generator, Issues of register allocation, Assignment and basic blocks, Code generation from DAG and the dynamic code generation algorithm

Unit 8 - Code Optimization **(8 Hrs)**

Sources of optimization, Peephole optimization and basic blocks, loops in flow graphs, Data flow analysis and equations, code improving transformation and aliases, Data flow analysis and algorithms, symbolic debugging of optimized code

---

Internal Continuous Assessment (ICA) :

ICA shall consist of minimum 10 practical assignment problems.

1. Generate the grammar for 'C' language.
2. Implement the lexical analyzer for simple 'C' language.
3. Implement the recognizer for given transition diagram.
4. Implement the top-down parsing using recursive decent parsing technique.
5. Implement the shift- reduce parser.
6. Implement the operator precedence parser.
7. Implement the LL(1) parser for the language.
8. Generate the symbol table for language.
9. Generation of 3- address code for language.
10. Implement the code optimization technique on the code produced in 10.
11. Generation of target code for given 3-address code.
12. Use of free open source software to practice the parsing example.

Text Books:

1. Compilers - Principles, Techniques and Tools, A.V. Aho, R. Shethi and J.D. Ullman (Pearson Education.)
2. Compiler Construction, Dhamdhere (Mc-Millan)
3. Principles of Complier Design- V.Raghavan (Mc Grawhill Education) (2nd edition)

Reference books:

1. Compiler Construction, Principles & Practice – Ken Louden (Cengage Learning)
2. Compiler Design in C,– Allen I. Holub (PHI / Pearson Education)
3. Compiler Construction: An advance course- Manish Kumar Jha (Dhanpat Rai) (3rd Edition)

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B. Tech (Computer Science & Engineering)
Semester – II
CS322 - UNIX Operating System

Teaching Scheme	**Examination Scheme**
Lectures: 3 Hrs/week, 3 Credits	ESE: 70 Marks
Practical: 2 Hrs/week, 1 Credit	ISE: 30 Marks
	ICA: 25 Marks
	POE : 50 Marks

Introduction :

Unix is the most powerful and popular multi-user and multi-tasking Operating System. In 1969 Ken Thompson, Dennis Ritchie, and others developed the basic building blocks of Unix including a hierarchical file system, i.e, the concepts of processes and a command line interpreter for the PDP-7. From there, multiple generations of Unix were developed for various machines.

Course Prerequisite:

Prerequisite knowledge of Operating Systems Concepts and 'C' programming language is needed.

Course Outcomes:

At the end of this course, the students will be able to,

1. Describe architecture of Unix, its kernel and file system.
2. Apply algorithms of regular file for inode assignment and disk block allocation.
3. Use system calls and program the Shell.
4. Describe structure of process, Memory and I/O management.
5. Implement programs using shell script.

SECTION–I

Unit 1 - Introduction (6 Hrs)

General Overview of the System- History, System Structure, User Perspective, Operating System Services, Assumption about Hardware

Introduction to KERNEL- Architecture of UNIX OS, Introduction to System Concepts

Unit 2 - The Buffer Cache (5 Hrs)

Buffer headers, Structure of the buffer pool, Scenarios for retrieval of a buffer, Reading and Writing disk blocks, Advantages and Disadvantages of cache.

Unit 3 - Internal Representation of Files (6 Hrs)

Inodes, Structure of the regular file, Directories, Conversion of a pathname to inode, Super block, Inode assignment to a new file, Allocation of disk blocks, Other file types.

Unit 4 - System Calls (6 Hrs)

Open, Read, Write, Lseek, Close, File Creation, Change directory and Change Root, Change Owner and Change mode, Stat and Fstat, Dup, Mounting and Unmounting, Link and Unlink

---

SECTION–II

Unit 5 - The Structure of process **(6 Hrs)**

Process states and transitions, Layout of system memory, The context of a process, Saving context of a process

Unit 6 - Process Control and Scheduling **(6 Hrs)**

Process Creation, Signals, Process Termination, Awaiting Process Termination, The user id of a process, System Boot and the init process, Process Scheduling, System call for time, Clock.

Unit 7 - Memory Management Policies **(5 Hrs)**

Swapping, Demand passing, a hybrid system with demand paging and swapping

Unit 8 - Shell Programming **(5 Hrs)**

Shell Variables: System Variables, Local Variables, Executing a Shell Script, The read command, Positional Parameters, The set command, Branching Control Structures (if, test command), Loop Control Structures (while, until, for), The expr Command

---

Internal Continuous Assessment (ICA) :

It should consist of minimum 8-10 experiments based on above topics. Following experiments may be conducted for the ICA.

Practical List:

1. Study of Unix architecture
2. Implement cp command
3. Implement ls command
4. Implement getblk algorithm
5. Implement ialloc & ifree algorithm.
6. Implement alloc and free algorithm.
7. Study of System calls
    - STAT & FSTAT,
    - PIPES,
    - LINK &UNLINK,
    - DUP,
    - MOUNT&UNMOUNT.
8. Study of shell programming
    - Find whether entered number is even or odd
    - Find factorial of number
    - Find whether entered number is prime or not
    - Print fibonnaci series
    - Find sum of series of entered number
    - Find power of number.
9. Implement malloc algorithm.
10. Study of KERNEL module programming

Text Books:

1. The design of Unix Operating Systems, Maurice J. Bach(PHI)
2. Introduction to UNIX & Shell Programming, M. G. Venkateshmurty (Pearson)
3. UNIX Concepts & Applications, Sumitabha Das

Reference Books:

1. Unix concepts and administration, Sumitabha Das, 3rd Edition (TMGH).
2. Unix Shell Programming, Yashwant Kanetkar, BPB Publications

---

Modalities for conducting End Semester Examination/Practical Oral Examination (POE)

- Practical and Oral Examination will be conducted by a panel of examiners assigned by university. A pair of examiners shall assess a batch @36 students in a day.
- The chairman shall prepare problem statements for a batch adhering to following guidelines:
  1. Problem statements shall be set in the context of course outcomes defined for the course.
  2. Minimum four problem statements shall be set for a batch.
  3. Problem Statements must be well described with no ambiguity. It should clearly mention the format of input and output along with constraints.
  4. Problem Statements shall be solvable in 2 hours by faculty and 2.5 hours by students.
  5. Problem statement shall be based on system calls, shell Programming, file, process and memory management algorithms

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B. Tech (Computer Science & Engineering)
Semester – II

CS323 - COMPUTER ORGANIZATION AND ARCHITECTURE

Teaching Scheme	**Examination Scheme**
Lectures: 3 Hrs/Week, 3 Credits	ESE: 70 Marks
	ISE: 30 Marks

Introduction: Computer Organization and Architecture (COA) course provide students with an understanding of the design of fundamental blocks used for building a computer system and interfacing techniques of these blocks to achieve different configurations of an "entire computer system". It introduces detailed understanding of various processor micro architectural designs, which include pipeline design, and multi-core processor design.

Course Prerequisite:
Student shall have undergone a course on Digital Logic Design and Operating system.

Course Outcomes:
At the end of the course students will be able to

1. Describe the functional architecture of computing systems.
2. Analyze various algorithms for arithmetic computation and arrive at fastest one.
3. Use ARC Processor based instructions to write assembly language program.
4. Demonstrate the design aspects of memory, instruction level parallelism and multiprocessors.

SECTION-I

Unit 1 - Introduction (4 Hrs)
A Brief History of Computing, The Von Neumann Model, Generations of Computers, The System Bus Model, Levels of Machines: Upward Compatibility, The Levels of computer, A Typical Computer System.

Unit 2 - Data Representation and Arithmetic (9 Hrs)
Introduction, Fixed Point Numbers, Floating Point Numbers, Fixed Point Addition and Subtraction, Fixed Point Multiplication and Division, Floating Point Arithmetic, High Performance Arithmetic: High Performance Addition, High Performance Multiplication, Introduction of arithmetic co-processor

Unit 3 - The Instruction Set Architecture and Memory (5 Hrs)
Hardware Components of the Instruction Set Architecture, ARC - A RISC Computer , Pseudo Operations, Synthetic Instructions, Examples of Assembly Language Programs, Accessing Data in Memory-Addressing Modes, The Memory Hierarchy, Cache Memory.

Unit 4 - Fundamentals of Computer Design (5 Hrs)
Introduction, Classes of Computers, Defining Computer Architecture, Trends in Technology, Power in Integrated Circuits and cost, Dependability, Measuring, Reporting and Summarizing Performance, Quantitative Principles of computer design.

SECTION-II

Unit 5 - Input/ Output Organization (4 Hrs)

External devices, I/O module, Programmed I/O, Interrupt driven I/ O, Direct memory access, I/O channels and processors, External interface.

Unit 6 - Fundamentals of Pipeline: (5 Hrs)

Introduction to Pipelining, The Major Hurdle of Pipelining: Pipeline Hazards, linear pipeline and Nonlinear pipeline, MESI protocol.

.

Unit 7 - Instructions –Level Parallelism (8 Hrs)

ILP: Concepts and challenges, Basic Compiler Techniques for exposing ILP, Reducing Branch costs with prediction, Overcoming Data hazards with Dynamic scheduling, Hardware based Speculation, Exploiting ILP using multiple issues and static scheduling,

Unit 8 - Multiprocessors and Thread –Level Parallelism: (5 Hrs)

Introduction, Symmetric Shared-Memory architectures, Performance of symmetric shared–memory multiprocessors, Distributed shared memory and Directory-based coherence.

Text Books:

1. Computer Architecture And Organization AN INTEGRAGATED APPROACH, Miles Murdocca and Vincent Heuring (WILEY).
2. Computer Architecture, A Quantitative Approach, John L. Hennessey and David A. Patterson: 4th Edition, Elsevier, 2007.
3. Computer Organization & Architecture, Rajaraman, PHI Learning

---

Reference books :

1. Computer Organisation, Hamacher Zaky (MGH)
2. Computer Architecture and Organization, John P. Hayes
3. Computer Organization and Architecture, William Stallings
4. Digital Logic and Computer Design, M. Morris Mano. Pearson Education - Prentice Hall

---

e-resource :

http://virtual-labs.ac.in/labs/cse10/cla_design.html or http://192.168.1.7:8080/vlab/

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B. Tech. (Computer Science & Engineering)
Semester – II
CS324 - ARTIFICIAL INTELLIGENCE

Teaching Scheme	**Examination Scheme**
Lectures– 3 Hrs/week, 3 Credits	ESE – 70 Marks
Practical – 2 Hrs/week, 1 Credit	ISE – 30 Marks
	ICA - 25 Marks

Introduction:

This course presents a basic introduction to the techniques used in developing Artificial Intelligent systems. It is a walkthrough to problem spaces and search algorithms, Knowledge representation, reasoning, logic programming and applications of Artificial Intelligence.

Course Prerequisite :

Student shall have some exposure to algorithms and programming.

Course Outcomes: At the end of the course students will be able to

1. Formulate and solve sequence of actions for an agent as a search problem.
2. Infer from represented knowledge using logical and probabilistic reasoning methods
3. Solve agent decision problems using probability theory
4. Comprehend forms of learning and demonstrate their working.

SECTION - I

Unit 1 – Overview **(6 hrs)**

Foundations, scope, problems, and approaches of AI.**Intelligent agents:** reactive, deliberative, goal-driven, utility-driven, and learning agents

Unit 2 - Problem-solving through Search **(7 hrs)**

Forward and backward, state-space, blind, heuristic, problem-reduction, A, A*, AO*, minimax, constraint propagation, neural, stochastic, and evolutionary search algorithms, sample applications.

Unit 3 - Knowledge Representation and Reasoning **(7 hrs)**

Ontologies, foundations of knowledge representation and reasoning, representing and reasoning about objects, relations, events, actions, time, and space; first order logic, situation calculus, description logics, reasoning with defaults, reasoning about knowledge, sample applications.

SECTION - II

Unit 4 - Representing and Reasoning with Uncertain Knowledge **(7 hrs)**

Probability, connection to logic, independence, Bayes rule, Bayesian networks, probabilistic inference, and sample applications.

Unit 5 - Decision-Making **(6 hrs)**

Basics of utility theory, decision theory, sequential decision problems, elementary game theory, sample applications.

Unit 6 - Learning and Knowledge Acquisition (7 hrs)

Forms of Learning: Supervised, Unsupervised, Semi-supervised. Statistical learning, Reinforcement Learning: Q-learning, sample applications.

Unit 7 - Conclusions (5 hrs)

Philosophical Foundations, AI: The Present and Future.

Internal Continuous Assessment (ICA):

ICA should consist minimum 10 assignments using any programming language with openly available tools, frameworks and resources based on the following topics.

- Intelligent agents
- Problem solving through search
- First order logic
- Bayesian Networks
- Decision and Game theory
- Statistical Learning
- Q-learning.

In addition to above students shall undertake a case study on **"Applications of AI: The Present and Future"**

Text Book:

1. Artificial Intelligence: A Modern Approach, Stuart Russell and Peter Norvig, 3rd Edition, Prentice Hall
2. A First Course in Artificial Intelligence, Deepak Khemani, McGraw Hill Education (India)
3. Introduction to Artificial Intelligence & Expert Systems, Dan W Patterson, PHI.

Reference Book:

Artificial Intelligence, Elaine Rich and Kevin Knight, Tata McGraw Hill

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B. Tech. (Computer Science & Engineering)
Semester – II
CS325 – MOBILE APPLICATION DEVELOPMENT

Teaching Scheme	**Examination Scheme**
Lectures– 2 Hrs./week, 2 Credits	ISE – 25 Marks
Practical – 2 Hrs./week, 1 Credit	ICA - 25 Marks
	POE – 50 Marks

Introduction: Mobile application development course will build your skills in creating mobile apps for Android platform as well as for Cross platform. This course includes Android application development and XamarinApplication development with basic User Interface design, basic building blocks, data handling, Testing mobile apps and how to take app to the market.

Course Prerequisite: Knowledge of programming paradigms and object-oriented programming principles.

Course Outcomes:

At the end of this course students will be able to

1. Select suitable development practices for a mobile application
2. Build cross platform mobile application for a given problem scenario.
3. Choose suitable method of testing, signing, packaging and distribution for a mobile application.

Section I : Mobile application development for Android using Java/Kotlin

Unit 1 - USER INTERFACE DESIGN (4 hrs)

Activity, Activity states, Activity Life Cycle, UI Resources, Layout Resources, String Resources, Image Resources, UI Elements and Events, Interaction between Activities, Exchanging data among activity, Fragments, Life Cycle of Fragments, Interaction between Fragments.

Unit 2 - MOBILE APPLICATION FUNCTIONALITY (4 hrs)

Beyond UI, App functionality beyond user interface - Threads, Asynchronous Tasks, Services – states and life cycle, Intent and Bound Service, Notifications, Intents and Intent Resolution, Broadcast receivers, Telephony and SMS APIs.

Unit 3 - NATIVE DATA HANDLING (4 hrs)

Native data handling On-device File I/O, data persistent and access using shared preferences, mobile databases such as SQLite and implementation for CRUD, and enterprise data access (via Internet/Intranet).

Unit 4 - TESTING & DISTRIBUTION OF MOBILE APPS (2 hrs)

Debugging mobile apps, White box testing, Black box testing, and Unit testing for Android. Versioning, signing and packaging mobile apps, distributing apps on mobile market place, Google play store.

Section II : Cross platform mobile application development using Xamarin

Unit 5 - C# PROGRAMMING GUIDE (4 hrs)

Inside a C# Program, Main() and Command-Line Arguments, Statements, Expressions, and Operators, Types, Classes and Structs, Interfaces, Inheritance and Polymorphism in C#, Delegates, Arrays, Strings, Properties, Indexers, Events, Generics, Iterators, Namespaces, Assemblies in .NET, Attributes, Collections, Exceptions and Exception Handling, Multi-threading.

Unit 6 - GETTING STARTED WITH XAMARIN (4 hrs)

Introducing native cross-platform applications with Xamarin, Hello MVVM—creating a simple cross-platform app using MVVM, MVVM—the model-view–view model design pattern, Hello again, MVVM—understanding and enhancing our simple MVVM app, An introduction to multithreading for Xamarin apps.

Unit 7 - BUILDING APPS (6 hrs)

Designing MVVM cross-platform apps, Building cross-platform models, Building cross-platform view models, Building simple Android views, Building more advanced Android views, Building simple iOS views, Building more advanced iOS views

Unit 8 - FROM WORKING CODE TO THE STORE (2 hrs)

Running mobile apps on physical devices, Testing mobile apps using Xamarin UITest, Using App Center to build, test, and monitor apps, Deploying apps to beta testers and the stores.

ISE Evaluation: ISE Evaluation for the course will consists of three hands on tests based on the topics mentioned in the syllabus.

Internal Continuous Assessment (ICA): Minimum 10 assignments requiring students to design, develop and test cross platform mobile applications for real world problem/use-case/scenario.

Text Books :

1. Android Application Development - All in one for Dummies, Barry Burd
2. Mobile Apps Development, Anubhav Pradhan, Anil V Deshpande
3. Xamarin in Action: Creating native cross-platform mobile apps by Jim Bennett, Manning Publications; 1st edition.

Reference books :

1. Android Developer Tools Essentials by Mike Wolfson (O'Reilly Media)
2. Embedded Android-Porting, Extending, and Customizing, Karim Yaghmour, (O'Reilly Media)

---

e-resources :

1. Android Developer Resources: http://developer.android.com
2. Xamarin documentation - Xamarin | Microsoft Docs: https://docs.microsoft.com/en-gb/xamarin/

Modalities for conducting End Semester Examination/Practical Oral Examination (POE):

- Practical and Oral Examination will be conducted by a panel of examiners assigned by university. A pair of examiners shall assess a batch @36 students in a day.
- The chairman shall prepare problem statements for a batch adhering to following guidelines:

1. At least Four Problem statements shall be set for a batch.
2. Problem statements shall be set in the context of course outcomes as defined in the course.
3. Problem Statements shall not be direct statements stating implement concept/topic etc.
4. Problem Statements shall be based on real world problem/use case/scenario etc.
5. Problem statement shall be at the minimum cognitive level of 'Apply' & above.
6. Problem Statements must be well described with no ambiguity and shall be of unseen nature.
7. Problem statements formulated shall be solvable by faculty in 2 hours and average students in 2.5 hrs in the examination duration.

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B. Tech. (Computer Science & Engineering)
Semester – II

CS326A - Elective-I : 1. OBJECT ORIENTED MODELING & DESIGN

Teaching Scheme	**Examination Scheme**
Lectures : 3 Hrs/week, 3 credits	ESE – 70 marks
Practical : 2 Hrs./week, 1 credit	ISE – 30 marks
	ICA – 25 marks

Introduction: This course presents Object Oriented approaches to software development based on modeling objects from the real world and then using the model to build a language independent design organized around those objects. These techniques promote better understanding of requirements, cleaner designs and more maintainable systems.

Course Prerequisite: Student shall have undergone an introductory course on Object oriented Design and programming

---

Course Outcomes:

At the end of the course students will be able to

1. Demonstrate basics of Object Oriented Modeling.
2. Design models for real world problems using Object Modeling Technique.
3. Design UML Diagrams for real world problems.

SECTION-I

Unit 1 - Introduction to Object oriented approach and Object Modeling (10 hrs)

Object Oriented development and themes, evidence for usefulness, modeling as a Design Technique. Objects, classes, links and associations, generalization and inheritance, grouping constructs, aggregation, abstract classes, generalization as extension and restriction, multiple inheritance, metadata, candidate keys and inheritance.

Unit 2 - Dynamic and Functional Modeling (6 hrs)

Events, states, operations, concurrency, nested state diagrams, advanced dynamic modeling concepts, relation of object and dynamic models, DFD, relation of functional to object and dynamic models

Unit 3 - Implementation of OMT (6 hrs)

Use of programming language and database system, Object oriented style, feature of object-oriented languages, Applications of OMT like object diagram compiler, Computer animation

SECTION-II

Unit 4 - Structural Modeling using UML (8 hrs)

Classes, Relationships, Common mechanisms. Diagrams, Class Diagrams, Interfaces, Types and Roles, Packages, Instances and Object Diagram.

Unit 5 - Behavioral Modeling using UML (8 hrs)

Interactions, Use cases, Use case diagram, Interaction Diagrams and Activity diagrams, Events and signals, State Machines, Processes and Threads, Time and space, State chart diagrams.

Unit 6 - Architectural Modeling using UML (7 hrs)

Components, Deployment, Collaboration, Patterns and Frame works, Component diagrams and Deployment Diagrams

---

Internal Continuous Assessment (ICA) :

ICA should consist of the following

1. Prepare a list of objects that you would expect each of the following system to handle also draw the class and object diagram for the same.
    a) Hospital Management System
    b) Air transportation system.
    c) Banking System
    d) Library Information System
    e) Railway Reservation System
    f) Water Management System
    g) Supermarket Information System
2. Dynamic and Functional Modeling
    a. Draw the state diagram for telephone answering machine. The machine should answer after five rings. If the telephone is answered before five rings, the machine should do nothing.
    b. Design functional model for flight simulator.
3. Draw Object Model with attributes and inheritance for Water Management System
4. Draw Use case Diagram for Railway Reservation System.
5. Draw Sequence and collaboration diagram for Banking System.
6. Draw Deployment diagram for Home Network. (Hint: Modern homes usually have a network of interconnected devices of different kinds and with various types of connections and communication protocols. It contains cable modem, wireless router, various computers and devices).
7. Draw Component diagram for online examination system.

Text Books:

1. Object Oriented Modeling and Design: Rambaugh, Premerlani, Eddy, Lorenson (PHI)
2. The Unified Modeling Language User Guide: Grady Booch, Jeams Rambaugh, Ivar Jacobson (Addison Wesley)
3. Object Oriented Modeling and Design with UML 2 Edition By Micheal R. Blaha, Pearson

---

Reference books :

1. Practical Object Oriented Design with UML, Mark Priestley.
2. UML-In a Nut Shell, Sinon Alhair

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B. Tech (Computer Science & Engineering)
Semester – II
CS326B - Elective-I : 2. ARTIFICIAL NEURAL NETWORK

Teaching Scheme	**Examination Scheme**
Lectures : 3 Hrs./week, 3 credits	ESE – 70 marks
Practical : 2 Hrs./week, 1 credit	ISE – 30 marks
	ICA – 25 marks

Introduction: This course introduces the basic models, learning algorithms and some applications of neural networks which will help to know how to use neural networks for solving various problems related to real world applications.

Course Prerequisite: Student shall have undergone a course on basic computational mathematics including matrices and probability.

Course Outcomes:

At the end of the course student will be able to

1. Demonstrate fundamentals of Artificial Neural Networks and their probable applications.
2. Select an appropriate configuration of neural network from those available to solve problems in a specific contest.
3. Demonstrate activation and synaptic dynamic models to understand stability and convergence in Artificial Neural Networks.
4. Develop applications of real world using Artificial Neural Networks.

SECTION – I

Unit 1 - Introduction to ANN (6 hrs)

Features, structure and working of Biological Neural Network, Trends in Computing Comparison of BNN and ANN

Unit 2 - Basics of Artificial Neural Networks (8 hrs)

History of neural network research, characteristics of neural networks terminology, models of neuron Mc Culloch – Pitts model, Perceptron, Adaline model, Basic learning laws, Topology of neural network architecture

Unit 3 - Back-propagation Networks (BPN) (8 hrs)

Architecture of feed forward network, single layer ANN, multilayer perceptron, back propagation learning, input - hidden and output layer computation, back-propagation algorithm, applications, selection of tuning parameters in BPN, Numbers of hidden nodes, learning.

SECTION – II

Unit 4 - Activation & Synaptic Dynamics (6 hrs)

Introduction, Activation Dynamics models, synaptic Dynamics models, stability and convergence, recall in neural networks.

Unit 5 - Basic functional units of ANN for pattern recognition tasks (8 hrs)

Basic feed-forward, Basic feedback and basic competitive learning neural network, Pattern association, pattern classification and pattern mapping tasks.

Unit 6 - Applications of ANN (8 hrs)

Pattern classification – Recognition of Olympic games symbols, Recognition of printed Characters. Neocognitron – Recognition of handwritten characters.

NET Talk: to convert English text to speech, Recognition of consonant vowel (CV) segments, texture classification and segmentation.

Internal Continuous Assessment (ICA) :

ICA should consist of following laboratory assignments

1. Introduction to Tensorflow, Graph-based computation
2. Training Neural Networks : Training Example Networks, Perceptrons, Shallow/Deep Networks
3. Constructing and Training Neural Networks : Working with Operators, Designing various training procedures
4. Convolutional Neural Networks : Training CNN, Examples from Computer Vision : Classification examples (AlexNet), Segmentation examples
5. Recurrent Neural Networks : Training RNNs Examples from NLP, Examples from Robotics

---

Text Books:

1. Artificial neural Networks, B. Yegnanarayana, PHI
2. Neural networks, Fuzzy logic and Genetic Algorithms, S.Rajsekaran, Vijayalakshmi Pari, PHI
3. Neural Networks, Satish Kumar

Reference Books:

1. Neural Networks a Comprehensive Foundations, Simon Haykin, PHI edition.
2. Neural Networks in Computer Intelligence, Li Min Fu TMH 2003
3. Neural Networks -James A Freeman David M S Kapura Pearson Education 2004.
4. Introduction to Artificial Neural Systems Jacek M. Zurada, JAICO Publishing House Ed. 2006

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B. Tech (Computer Science & Engineering)
Semester – II
CS326C - Elective-I : 3. DATA SCIENCE

Teaching Scheme	**Examination Scheme**
Lectures : 3 Hrs./week, 3 credits	ESE – 70 marks
Practical : 2 Hrs./week, 1 credit	ISE – 30 marks
	ICA – 25 marks

Introduction:

Data science is a field of study and application that has been growing rapidly for the past several decades. As a growing field, it is gaining a lot of attention in both the media as well as in the job market. This course introduces the basic terminology used by data scientists and a look at the types of problem.

Course Prerequisite : Introduction to Programming, Probability

---

Course Outcome:

At end of this course, the students will be able to:

1. Demonstrate understanding of the mathematical foundations needed for data science.
2. Collect, explore, clean and manipulate data to convert it into an appropriate form .
3. Implement models such as k-nearest Neighbors, Naive Bayes, linear and logistic regression, decision trees, neural networks and clustering.
4. Build data science applications using Python based toolkits.

SECTION – I

Unit 1 - Introduction to Data Science **(4 hrs)**

Concept of Data Science, Traits of Big data, Web Scraping, Analysis vs Reporting

Unit 2 - Introduction to Programming Tools for Data Science **(6 hrs)**

Toolkits using Python: Matplotlib, NumPy, Scikit-learn, NLTK
Visualizing Data: Bar Charts, Line Charts, Scatterplots
Working with data: Reading Files, Scraping the Web, Using APIs (Example: Using the Twitter APIs), Cleaning and Munging, Manipulating Data, Rescaling, Dimensionality Reduction

Unit 3 - Mathematical Foundations **(12 hrs)**

Linear Algebra: Vectors, Matrices
Statistics: Describing a Single Set of Data, Correlation, Simpson's Paradox, Correlation and Causation, Probability: Dependence and Independence, Conditional Probability, Bayes's Theorem, Random Variables, Continuous Distributions, The Normal Distribution, The Central Limit Theorem Hypothesis and Inference: Statistical Hypothesis Testing, Confidence Intervals, P- hacking, Bayesian Inference

SECTION – II

Unit 4 - Machine Learning **(6 hrs)**

Overview of Machine learning concepts – Over fitting and train/test splits, Types of Machine learning – Supervised, Unsupervised, Reinforced learning, Introduction to Bayes Theorem, Linear Regression- model assumptions, regularization (lasso, ridge, elastic net)

Unit 5 - Machine Learning Algorithms **(10 hrs)**

Classification and Regression algorithms- Naïve Bayes, K-Nearest Neighbors, logistic regression, support vector machines (SVM), decision trees, and random forest, Classification Errors, Analysis of Time Series- Linear Systems Analysis, Nonlinear Dynamics, Rule Induction, Neural Networks- Learning And Generalization, Overview of Deep Learning.

Unit 6 - Case Studies of Data Science Application **(6 hrs.)**

Weather forecasting, Stock market prediction, Object recognition, Real Time Sentiment Analysis.

Internal Continuous Assessment (ICA) :

ICA should consist of writing programs in Python for the following

1. To predict the class of the flower based on available attributes.
2. To predict if a loan will get approved or not.
3. To predict the traffic on a new mode of transport.
4. To predict the class of user.
5. To indentify the tweets which are hate tweets and which are not.
6. To predict the age of the actors.
7. To predict the time taken to solve a problem given the current status of the user.

Text Books :

1. Data Science from Scratch: First Principles with Python, Joel Grus, O'Reilly Media
2. Data Sciences, Jain V.K., Khanna Publishing House, Delhi.
3. Machine Learning, Jeeva Jose, Khanna Publishing House, Delhi.

Reference Books :

1. Hands-On Machine Learning with Scikit-Learn and Tensor Flow: Concepts, Tools, and Techniques to Build Intelligent Systems - Aurélien Géron, 1st Edition, O'Reilly Media
2. Big Data and Hadoop, Jain V.K., Khanna Publishing House, Delhi.
3. Machine Learning, Chopra Rajiv, Khanna Publishing House, Delhi.
4. Deep Learning, Ian Goodfellow, Yoshua Bengio and Aaron Courville, MIT Press http://www.deeplearningbook.org
5. Data Mining Concepts and Techniques, Jiawei Han and Jian Pei Third Edition, Morgan Kaufmann Publishers

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B. Tech (Computer Science & Engineering)
Semester – II
CS327 : Mini Project

Teaching Scheme	**Examination Scheme**
Practical: 2 Hrs/week, 1 Credit	ICA : 25 Marks
	POE – 50 marks

Course Outcomes :

At the end of this course, student will be able to

1. Select mini project problem of societal relevance in selected domain
2. Design system architecture with due consideration of environment, sustainability and ethics.
3. Develop the solution to the problem using tools, resources and frameworks.
4. Engage in teamwork and communicate effectively, while observing professional ethics.
5. Inculcate habit of self study and lifelong learning.

Note :

1. There should be a group of preferably 4/5 students.
2. Students should be given projects in Hardware, Software, Embedded or any contemporary topic.
3. One guide should be allocated per group.

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B. Tech (Computer Science & Engineering)
Semester – II
Self Learning Module II (Technical)
SL32A. UI / UX Technology

Examination Scheme
2 Credits **ESE - 50 marks**

Course Prerequisite:

Student should have knowledge of basic programming. They should also have basic knowledge of GUI and its working.

Course Outcomes:

1. Design, develop and apply styling to a web-based application.
2. To be able to design responsive web design.
3. Build efficient and scalable web API and application.
4. Develop lightweight browser based functionalities leveraging client side scripting framework.

SECTION -I

Unit 1 - HTML5, CSS3, DHTML

HTML: HTML Review, HTML5 Introduction, Futures of HTML5, New elements in HTML5 (canvas, Media elements, Form elements, Semantic and structural elements,

CSS: CSS Introduction ,CSS Syntax, CSS Id & Class, CSS Styling, CSS3: Selectors, Box Model, Backgrounds and Borders, Image Values and Replaced Content, Text Effects, 2D/3D Transformations, Animations, Multiple Column Layout, User Interface. DHTML

Unit 2 - JavaScript and jQuery

Introduction, Understanding of variables, data types, control flow, and basic function usage in JavaScript, Event Handling, JS Built-in Objects

JSON: JavaScript Object and Array Creation Using Literals, JavaScript Objects in Arrays & Arrays in Objects, JSON syntax, JSON Parsers, JSON Data Transfer Between Client and Server, AJAX.
jQuery: jQuery Fundamentals, using jQuery Selectors, Interacting with the DOM, Handling Events

SECTION -II

Unit 3 - Responsive Web Design

What is Responsive Design, Responsive Content, HTML for Responsive Sites, CSS for Responsive Sites, Media Queries, Images, Responsive Workflow, Mobile and Beyond, Typography, Navigation and Header Layout. Introduction to Bootstrap, Bootstrap CSS, Bootstrap Layout Components, Bootstrap JavaScript Plugins

Unit 4 - Dreamweaver & PSD to HTML conversion

Dreamweaver Basics, Dreamweaver shortcuts, Dreamweaver panels, Dreamweaver toolbars, Dreamweaver Automation, Source code format.

Unit 5 - PSD to HTML conversion: Source code formatting Reading the design, PSD to HTML conversion,

Dynamic content logic, Cross browser compatibility, slicing the PSDs, Image optimization, setting up Naming convention.

Text Book :

1. Head First HTML5 Programming by Eric Freeman, Elisabeth Robson O"Reilly Media
2. Designing Next Generation Web Projects with CSS3 by Sandro Paganotti- CreateSpace Independent Publishing Platform.
3. JavaScript, A Geginners Guide, Third Edition by John Pollock-McGraw_Hill Osborne Media.
4. Adobe Dreamweaver CC Classroom in a Book By Jim Maivald
5. Bootstrap by Jake Spurlock, Published by O"Reilly Media

Reference book :

1. HTML5 Black Book: Covers CSS3, Javascript, XML, XHTML, Ajax, PHP and Jquery by Kogent Learning Solutions Inc.
2. HTML5 Application, Zachary Kessin, O"Reilly, Shroff Publishing and Distributions Pvt. Ltd.
3. Responsive Design Workflow –Stephen N. Hay.

PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR
Faculty of Science and Technology
Third Year B. Tech (Computer Science & Engineering)
Semester – II
Self Learning Module II (Technical)
SL32B. Software Licences and Practices

Examination Scheme
2 Credits **ESE - 50 marks**

Course Outcomes:

1. Students will develop basic understanding of software licensing models and practices adopted in software development and distribution.
2. Students will be able to analyze and choose appropriate software licensing model and strategy for their own softwares developed.

Unit 1 - Introduction

Basic Principles of Copyright Law, Contract and Copyright, Open Source Software Licensing, Issues, with Copyrights and Patents, The Open Source Definition, Warranties.

Unit 2 - Software Licenses

The MIT License, The BSD License , The Apache License, v1.1 and v2.0, The Academic Free License, Application and Philosophy of MIT and BSD Licenses, GNU General Public License, GNU Lesser General Public License, The Mozilla Public License, Application and Philosophy of GNU GPL and GNU LGPL.

Unit 3 - Creative Commons Licenses and Non Open Source Software Licenses

Creative Commons Licenses, Classic Proprietary License, Sun CommUnity Source License, Microsoft Shared Source Initiative.

Unit 4 - Legal Impacts of Open Source and Free Software Licensing

Entering Contracts, Statutory Developments Related to Software Contracts, The Self Enforcing Nature of Open Source and Free Software Licenses, The Global Scope of Open Source and Free Software Licensing, The "Negative Effects" of Open Source and Free Software Licensing, CommUnity Enforcement of Open Source and Free Software Licenses, Compatible and Incompatible Licensing: Multiple and Cross Licensing.

Text Books:

1. Intellectual Property and Open Source: A Practical Guide to Protecting Code - By Van Lindberg, Oreily Media.
2. Understanding Open Source and Free Software Licensing - By Andrew M. St. Laurent, Oreily Media. (e-Resource available at : http://oreilly.com/openbook/osfreesoft/book/index.html)

Reference Books:

1. Essentials of Licensing Intellectual Property - By Alexander I. Poltorak and Paul J. Lerner, John Wiley Publication.
