MAR IVANIOS COLLEGE (AUTONOMOUS)
THIRUVANANTHAPURAM

REGULATION, SCHEME & SYLLABUS

FOR

BSc. COMPUTER SCIENCE

(UNDER CBCS CAREER RELATED 2(b) SCHEME)

(2019 ADMISSION ONWARDS)

INTRODUCTION

Information Technology has today become an integral part of all industry domains as well as fields of academics and research. Bachelor of Computer Science is a 3-year under graduate programme spread over 6 semesters. The programme is designed to bridge the gap between IT industries and academic institutes by incorporating the latest developments into the curriculum and to give students a complete understanding within a structured framework. A real genuine attempt has been made while designing the new syllabus for this 3-year under graduate programme. It prepares the students for a career in computer/information technology and also motivates them towards further studies and research opportunities. The curriculum supports students to gain adequate programming practices along with theoretical foundation and also includes interdisciplinary courses and electives for widening the domain expertise.

AIMS AND OBJECTIVES

- To provide a sound understanding of computer programming to students for a perfect start for a challenging career in the field of information technology.
- To help students apply the knowledge acquired in domains like Mathematics, Professional Subjects and Environmental Issues.
- To train students develop strong foundation in theoretical and experimental work for being able to analyze, synthesize and design products, processes and systems as desired.
- To cater to the ever changing demands of information technology along with necessary inputs from the industry.
- To inculcate social awareness, logical thinking, communication skills & professionalism among to work as a team in maintaining diverse environment towards the betterment of society.

REGULATION

ELIGIBILITY FOR ADMISSION

Eligibility for admissions and reservation of seats for BSc Computer Science Programme shall be in accordance with the rules framed by the University/Institution from time to time. Eligibility for admission is a pass in Higher Secondary or any other examination recognized as equivalent thereto by the University of Kerala with Mathematics as one of the optional subjects. The index marks will be the total marks obtained for the qualifying examination plus marks for Mathematics and Physics/Computer Science. For the candidates who have studied Computer Science at +2 level, the score obtained in Computer Science will be added, instead of the score in Physics (2011 Admission onwards – U.O. No.Ac.A.IV/3150748/2011 dtd. 10.06.2011).

DURATION

The duration of BSc Computer Science Programme shall be three years consisting of six semesters. The duration of each semester shall be five months inclusive of the days of examinations. There shall be at least 90 working days in a semester.

CREDIT CALCULATION

The following formula is used for conversion of credit into hours.

- One Credit would mean equivalent of 18 periods of 60 minutes each, for theory, workshops/labs and tutorials.

GENERAL COURSE STRUCTURE

Study Components	Semester	Number of courses	Credits/ Course	Total Credits
Language course: English	I & II	2	2	4
Foundation Courses	I	2	3	6
Core Course	All Semesters	20-32	2-4	74-98
Complementary Courses	I & II	0-4	2-3	0-12
Open Course	V	1	2	2
Elective course	V & VI	1-6	2-4	2-22
Project/Dissertation	V & VI	1-2	2-4	6

Accumulated Total Minimum Credits required for Programmes of study = 120 Credits.

Minimum Duration = 6 Semesters

SCHEME

Semester I:

Sl. No.	Course code	Course Name	Credits	Hours per week			
				Lecture	Tutorial	Lab	Total
Language Course							
1.	AUEN111.5	Language Skills	**2**	**3**	**-**	**-**	**3**
Complementary Course							
2.	AUMM 131.2f	MathematicsI: Calculus and Number Theory	**3**	**3**	**1**	**-**	**4**
Foundation Courses							
3.	AUCS 121	Computer Fundamentals and Organization	**3**	**3**	**1**	**-**	**4**
4.	AUCS122	Digital Electronics	**3**	**3**	**-**	**-**	**3**
Core Courses							
5.	AUCS141	C Programming	**3**	**3**	**-**	**-**	**3**
6.	AUCS14PI	C Programming Lab	**3**	**-**	**-**	**4**	**4**
7.	AUCS14PII	IT Lab	**3**	**-**	**1**	**3**	**4**
		TOTAL	**20**	**15**	**3**	**7**	**25**

Semester II

Sl. No.	Course code	Course Name	Credits	Hours per week			
				Lecture	Tutorial	Lab	Total
Language Course							
1.	AUEN211.5	English for Career	**2**	**3**	**-**	**-**	**3**
Complementary Course							
2.	AUST231.2f	MathematicsII:Discrete Mathematics	**3**	**3**	**1**	**-**	**4**
Core Courses							
3.	AUCS 241	Environmental Studies	**3**	**3**	**-**	**-**	**3**
4.	AUCS 242	Data Structures using C	**3**	**3**	**1**	**-**	**4**
5.	AUCS 243	Web Programming	**3**	**3**	**-**	**-**	**3**
6.	AUCS 24PI	Data Structures using C Lab	**3**	**-**	**-**	**4**	**4**
7.	AUCS 24PII	Web Programming Lab	**3**	**-**	**-**	**4**	**4**
		TOTAL	**20**	**15**	**2**	**8**	**25**

Semester III

Sl. No.	Course code	Course Name	Credits	Hours per week			
				Lecture	Tutorial	Lab	Total
Core Courses							
1.	AUCS 341	Programming in C++	**3**	**3**	**-**	**-**	**3**
2.	AUCS 342	Data Communication & Computer Networks	**3**	**3**	**1**	**-**	**4**
3.	AUCS 343	System Software	**3**	**3**	**-**	**-**	**3**
4.	AUCS 344	Value Education	**2**	**3**	**-**	**-**	**3**
5.	AUCS 345	Database Management Systems	**3**	**3**	**1**	**-**	**4**
6.	AUCS 34PI	Programming in C++ Lab	**3**	**-**	**-**	**4**	**4**
7.	AUCS 34PII	DBMS Lab	**3**	**-**	**-**	**4**	**4**
		TOTAL	**20**	**15**	**2**	**8**	**25**

Semester IV

Sl. No.	Course code	Course Name	Credits	Hours per week			
				Lecture	Tutorial	Lab	Total
Core Courses							
1.	AUCS 441	Operating Systems	**3**	**3**	**1**	**-**	**4**
2.	AUCS 442	Software Engineering	**3**	**3**	**1**	**-**	**4**
3.	AUCS 443	Internet of Things	**3**	**3**	**-**	**-**	**3**
4.	AUCS 444	PHP	**3**	**3**	**-**	**-**	**3**
5.	AUCS 445	Programming in Java	**3**	**3**	**-**	**-**	**3**
6.	AUCS 44PI	Programming in Java Lab	**3**	**-**	**-**	**4**	**4**
7.	AUCS 44PII	PHP Lab	**3**			**4**	**4**
		TOTAL	**21**	**15**	**2**	**8**	**25**

Semester V

Sl. No.	Course code	Course Name	Credits	Hours per week			
				Lecture	Tutorial	Lab	Total
Core Courses							
1.	AUCS 541	Computer Graphics	**3**	**3**	**-**	**-**	**3**
2.	AUCS 542	Design & Analysis of Algorithms	**3**	**3**	**1**	**-**	**4**
3.	AUCS 543	Python Programming	**3**	**3**	**-**	**-**	**3**
4.	AUCS 54PI	Python Programming Lab	**3**	**-**	**-**	**4**	**4**
5.	AUCS 544	Major Project Phase I	**2**	**-**	**-**	**4**	**4**
Open Courses			**2**	**3**	**-**	**-**	**3**
6.	AUCS 581.a	Cyber Security					
	AUCS 581.b	Digital Marketing					
	AUCS 581.c	Internet And WWW					
Elective Courses			**3**	**3**	**1**	**-**	**4**
7.	AUCS 591.a	Data Analytics					
	AUCS 591.b	MobileComputing					
	AUCS 591.c	Trends in Computing					
		TOTAL	**19**	**15**	**2**	**8**	**25**

Semester VI

Sl. No.	Course code	Course Name	Credits	Hours per week			
				Lecture	Tutorial	Lab	Total
Core Courses							
1.	AUCS641	Data Mining And Warehousing	**4**	**4**	**-**	**-**	**4**
2.	AUCS642	Information Security	**4**	**4**	**-**	**-**	**4**
3.	AUCS643	Artificial Intelligence	**4**	**3**	**1**	**-**	**4**
4.	AUCS644	Major Project Phase II	**4**	**-**	**-**	**9**	**9**
Elective Courses			**4**	**4**	**-**	**-**	**4**
5.	AUCS691.a	Computational Linguistics					
	AUCS691.b	Cyber Forensics					
	AUCS691.c	Free and Open Source Software					
		TOTAL	**20**	**15**	**1**	**9**	**25**

DISTRIBUTION OF MARKS

Course Code	Subject	Credit	Contact Hrs	Marks ESE	Marks CE	Total Marks
SEMESTER I						
AUEN111.5	Language Skills	2	54	80	20	100
AUMM131.2f	Mathematics I: Calculus and Number Theory	3	72	80	20	100
AUCS 121	Computer Fundamentals and Organization	3	72	80	20	100
AUCS 122	Digital Electronics	3	54	80	20	100
AUCS141	C Programming	3	54	80	20	100
AUCS14PI	C Programming Lab	3	72	80	20	100
AUCS14PII	IT Lab	3	72	80	20	100
SEMESTER II						
AUEN211.5	English for Career	2	54	80	20	100
AUMM231.2f	Mathematics II: Discrete Mathematics	3	72	80	20	100
AUCS 241	Environmental Studies	3	54	80	20	100
AUCS 242	Data Structures using C	3	72	80	20	100
AUCS 243	Web Programming	3	54	80	20	100
AUCS 24PI	Data Structures using C Lab	3	72	80	20	100
AUCS 24PII	Web Programming Lab	3	72	80	20	100
SEMESTER III						
AUCS 341	Programming in C++	3	54	80	20	100
AUCS 342	Data Communication & ComputerNetworks	3	72	80	20	100
AUCS 343	System Software	3	54	80	20	100
AUCS 344	Value Education	2	54	80	20	100
AUCS 345	Database Management Systems	3	72	80	20	100
AUCS 34PI	Programming in C++ Lab	3	72	80	20	100
AUCS 34PII	DBMS Lab	3	72	80	20	100
SEMESTER IV						
AUCS 441	Operating Systems	3	72	80	20	100
AUCS 442	Software Engineering	3	72	80	20	100
AUCS 443	Internet of Things	3	54	80	20	100
AUCS 444	PHP	3	54	80	20	100
AUCS 445	Programming in Java	3	54	80	20	100
AUCS 44PI	Programming in Java Lab	3	72	80	20	100
AUCS 44PII	PHP Lab	3	72	80	20	100

Code No	Subject	Credit	Contact Hrs	Marks ESE	Marks CE	Total Marks
SEMESTER V						
AUCS 541	Computer Graphics	3	54	80	20	100
AUCS 542	Design & Analysis of Algorithms	3	72	80	20	100
AUCS 543	Python Programming	3	54	80	20	100
AUCS 54PI	Python Programming Lab	3	72	80	20	100
AUCS 544	Major Project Phase I	2	72	80	20	100
Open Courses		2	54	80	20	100
AUCS 581.a	Cyber Security					
AUCS 581.b	Digital Marketing					
AUCS 581.c	Internet And WWW					
Elective Courses		3	72	80	20	100
AUCS 591.a	Data Analytics					
AUCS 591.b	Mobile Computing					
AUCS 591.c	Trends in Computing					
SEMESTER VI						
AUCS641	Data Mining And Warehousing	4	72	80	20	100
AUCS642	Information Security	4	72	80	20	100
AUCS643	Artificial Intelligence	4	72	80	20	100
AUCS644	Major Project Phase II	4	162	80	20	100
Elective Courses		4	72	80	20	100
AUCS691.a	Computational Linguistics					
AUCS691.b	Cyber Forensics					
AUCS691.c	Free and Open Source Software					
	TOTAL MARKS	**120**		**3200**	**800**	**4000**

DISTRIBUTION OF MARKS (LAB EXAMINATION)

1. Program in Part A should be sufficiently simple – 25 marks (Logic – 10 marks, Successful compilation – 10 marks, Result – 5 marks)
2. Program in Part B should be based on advanced concepts - 30 marks (Logic – 15 marks, Successful compilation – 10 marks, result – 5 marks)
3. Viva Voce - 15 marks
4. Lab Record - 10 marks

Total Marks - 80 marks

SOCIAL SERVICE/ EXTENSION ACTIVITIES

Students are to participate in Extension/ NSS/ NCC or other specified social service, sports, literary and cultural activities at least for a minimum of 20 hours each during 3rd/4thsemester. These activities have to be carried out outside the Instructional Hours and are mandatory for all students.

ATTENDANCE

Students who secure a minimum of 75% attendance in the aggregate for all the Courses of a semester taken together alone will be allowed to register for End Semester Evaluation. Others have to repeat the semester along with the next batch, unless they could make up the shortage of attendance through condonation. However the award of Grade for attendance in CE shall be made course wise. Condonation of shortage of attendance to a maximum of 10days in a semester subject to a maximum of two times during the whole period of a Degree Programme shall be granted by the College on valid grounds. This condonation shall not be considered for awarding marks for CE. Benefits of attendance for a maximum of 10 days in a semester shall be granted to students who participate/attend University Youth festivals, meetings of the UniversityBodies and Extra Curricular Activities (NCC/NSS/SPORTS), on production of participation/attendance certificate by the concerned University/College faculty in charge as the case may be. But in such cases, condonation will be considered for award of marks for CE.

EVALUATION

There shall be Continuous Evaluation (CE) and End Semester Evaluation (ESE) for BSc Computer Science Programme. CE is based on specific components viz., attendance, tests, assignments and seminars. The CE shall carry a weightage of 20 per cent and ESE shall carry a weightage of 80 per cent.

Attendance(Max. Marks: 5)

The allotment of marks for attendance shall be as follows:

Attendance (in %)	Mark (out of 5)
> 90	5
>85 & ≤90	4
>80 & ≤85	3
>75 & ≤80	2
75	1
< 75	0

Assignments or Seminars (Max. Marks:5)

Each student shall be required to do one assignment or one seminar foreach Course. Valued assignments shall be returned to the students. The seminars shall be organized by the teacher/teachers in charge of CE and the same shall be assessed by a group of teachers including the teacher/ teachers in charge of that Course. Assignments/Seminars shall be evaluated on the basis of their quality. The teacher shall define the expected quality of an assignment in terms ofstructure, content, presentation etc. and inform the same to the students. Due weight shall be given for punctuality in submission. Seminar shall be similarly evaluated in terms of structure, content, presentation, interaction etc.

Test (Max. Marks: 10)

For each Course there shall be one class test during a semester. Valued answer scripts shall be made available to the students for perusal within 15 working days from the date of the test.

END SEMESTER EVALUATION (ESE)

End Semester Examination of all the courses in all semesters shall be conducted. The duration of examination of all courses shall be 3 hours.

ANNOUNCEMENT OF RESULTS OF CE

The results of the CE shall be displayed within 5 working days from the last day of a semester. Complaints regarding the award of marks for CE if any, have to be submitted to the Head of the Department within 3 working days from the display of results of CE. These complaints shall be examined by the DLMC and shall arrive at a decision, which shall be communicated to the student. The Statement of marks of the CE of all the students shall be approved bythe Department Committee, countersigned by the Principal and forwarded to the Controller of Examinations within 15 working days from the last day of the semester.

EVALUATION OF PROJECT

The report of the project shall be submitted to the Department in duplicate before the completion of the semester and are to be produced before the examiners appointed by the institution. A board of two examiners appointed by the institution shall evaluate the report of the project work. The viva – voce based on the project report shall be conducted individually.

GRADING

Both CE and ESE will be carried out using direct grading system on a 7 point scale and the grades are given below:

Criteria for Grading:

Grade Range	Grade	Performance
9 and above	A+	Outstanding
8 to <9	A	Excellent
7 to <8	B	Very Good
6 to <7	C	Good
5 to <6	D	Satisfactory
4 to <5	E	Adequate
<4	F	Failure

PROMOTION TO HIGHER SEMESTERS

Students who complete the semester by securing the minimum required attendance and by registering for the End Semester Examination of each semester conducted by the College alone shall be promoted to the next higher semester.

PASS CRITERIA

An aggregate minimum of 40% marks is required for passing a course with aseparate minimum of 35% marks for CE and ESE. For the successful completion of a semester a student has to score a minimum SCPA of 4. For the successful completion of a programme and award of degree, a student must pass all courses satisfying the minimum credit requirement and must score a minimum CCPA of 4 and an overall grade of E.

RE-APPEARANCE OF FAILED STUDENTS

Students who fail shall have to reappear for the ESE of the same along with the next regular batch of students. The number of chances or such appearances is limited to 5 and the same have to be done within a period of 12 continuous semesters including the semester in which they have first appeared.

SYLLABUS

SEMESTER I

LANGUAGE COURSE I

AUEN111.5: LANGUAGESKILLS

Credit: 2 **Instructional Hours: 3 per week**

LEARNING OBJECTIVES:

1. Mastering the language for personal and professional growth.
2. Basic language skills are to be acquired through interactive classroom sessions
3. Connecting literature with language learning

LEARNING OUTCOMES:

English as an acquired language for undergraduate students is to be mastered with focus on learning the basic skills of listening, speaking, reading and writing the language proficiently. This course aims to impart these skills in an interactive manner along with classroom activities and using the text as a resource for self-study as well. Discursive Practice as the learning and teaching method for this course, will encourage teachers to localise and personalise learning of English for students in undergraduate classrooms. The course will equip the students with basic language skills along with improved non-verbal skills thereby improving their employability quotient.

COURSE DESCRIPTION

Module I: Basics of Communication

UNIT I

Theories of communication - Types of communication - Effective communication - barriers to effective communication - English as a language for communication - Micro-skills and macro-skills

Module II: Listening

Unit II

Lesson I – Listening
Lesson II – Short Story
Lesson III – Poem
Lesson IV – Activities

1. Short Story: "The Night Train at Deoli" by Ruskin Bond – Comprehension questions
2. Poem: "Lines Addressed to a Warrior" by Meena Kandasamy -Comprehension questions

Module III: Speaking

Unit IV

Lesson I – Speaking
Lesson II – Speech
Lesson III – Poem
Lesson IV – Activities

Speaking Skills - enhancing speaking skills - public speaking - telephonic conversations - podcasting – anchoring

1. Speech by Steve Jobs – Comprehension Questions
2. Poem - "Still I Rise" by Maya Angelou – Comprehension Questions

Module IV: Reading

Unit VI

Lesson I –Reading
Lesson II –Autobiography
Lesson III –Essay
Lesson IV – Activities

1. *Autobiography* (Excerpt) Nelson Mandela – Comprehension Questions
2. "Toba TekSingh" by Sadat Hasan Manto– Comprehension Questions

Module V: Writing

Unit IX

Lesson I – Writing
Lesson II – Essay
Lesson III – Blog
Lesson IV – Activities

Module VI: Soft Skills

Unit X

Lesson I – Soft Skills

Posture - gestures - eye contact - telephone etiquette - netiquette - interpersonal skill.

Text Book Prescribed:

Language Skills: A Course on Communication Skills in English

By Dr.Swapna Gopinath, Associate Professor, Department of English, S.N College, Chempazhanthy and Sangeetha Hariharan, Assistant Professor, Department of English, S.N Women's College, Kollam.

Publishers: Emerald

Suggested Reading

S .P. Dhanvel. *English and Softskills*. Orient Blackswan, 2010.
Dr M. Farook. *English for Communication*, Emerald Publishers, 2015. Dr
Mathew Joseph. *Fine-tune your English*. Orient Blackswan, 2010.
E. Suresh Kumar, B Yadava Raju and C Muralikrishna. *Skills in English*. Orient Blackswan, 2013.
Bill Bryson. *The Mother Tongue: English and How it Got it that Way*. Harper Collins, 1990.

Web sources

www.englishclub.com
http://www.bbc.co.uk/learningenglish/
https://www.eslfast.com/
https://www.myenglishpages.com/
http://www.examenglish.com/
http://learnenglishteens.britishcouncil.org/exams/listening-exams
https://www.cambridgeenglish.org/learning-english/
https://www.pearson.com/us/

Model Question Paper

Question paper setter, please note that questions are NOT to be asked from the passages and poems given in this text. They are meant to help in learning the basic language skills. Internal exams should focus on listening and speaking skills. Writing and reading skills will be tested as part of End Semester Examinations

No questions should be asked from Additional/ Suggested Reading

Hours: 3 **80 marks**

Section A

Answer in a word or a sentence. All Questions carry one mark each

Questions 1 to 10 will be on suprasegmental features and will focus on words and simple sentences.

(10 x 1 = 10marks)

Section B

Questions based on the theories of communication

Attempt Eight out of Twelve Short answers picked from first and sixth module

(8 x 2 = 16 marks)

Section C

Attempt Six out of nine questions to be answered in around 100 words

Questions based on the activities from all the modules except Module I and Module II

(6 x 4 = 24 marks)

Create a dialogue,Write a script for anchoring ,Write a speech, Write a telephone conversation Write a blog, Write a script for a podcast, Write a formal/informal letter, Edit the passage given below, Write an Email, Write minutes for a meeting, Write a report, Edit the passage

Section D

Questions based on the activities from all the modules except Module I and Module II Attempt any two out of four questions in about 300 words

(15 x2 = 30 marks)

Write an essay on any one of the topics
Write a paragraph on two of the following

COMPLEMENTARY COURSE I

AUMM131.2f - MATHEMATICS 1: CALCULUS AND NUMBER THEORY

Credit: 3 **Instructional Hours: 4 per Week**

MODULE1

DIFFERERENTIATION ANDITS APPLICATIONS **18Hours**

Differentiation: Hyperbolic and inverse hyperbolic functions.

Applications:*n*th-derivative of-polynomials,exponential,sine,cosine and their product, Leibnitz Theorem (Without Proof) and its application

The topics in this module can be found in Chapter 6, sections 6.7 **and** 6.8 **of Text [1] and Chapter 4, sections** 4.1 **and** 4.2 **of Text [2].**

MODULE 2

INTEGRATION AND ITS APLLICATIONS **18Hours**

Definite and Indefinite Integrals: Integration techniques - substitution, rational functions with degree of numerator less than and greater than or equal to the degree of denominator, partial fraction and integration by parts.

Applications: Area of a curve, area between two curves, length of a plane curve, area of a surface of revolution and volume of revolution.(volume by cylindrical shells, volume by slicing are excluded)

The topics in this module can be found in Chapter 4, sections 4.1**,** 4.2**,** 4.3 **and** 4.5 **and Chapter 5, sections** 5.1**,** 5.4 **and** 5.5 **of Text [1].**

MODULE 3

ORDINARY DIFFERENTIALEQUATIONS **18Hours**

Ordinary Differential Equations: Solution of Higher Order differential equations withconstant coefficients (homogeneous and non-homogeneous - exponential, sine, cosine and hyperbolic functions and their combinations), solution of first and second order simultaneous system of equations, Cauchy-Euler type differential equations, Legendre's differential equations.

The topics in this module can be found in Chapter 5, sections 5.2, 5.3, 5.7 and 5.8 and Chapter 7, section 7.2 of Text [3].

MODULE 4

Number Theory **18Hours**

Numbers: Euclid's Algorithm - GCD of 2 natural numbers, Divisors of a given natural number,Highest power of a prime.

Congruences: Euler's function φ(n) and its properties(without proof of theorems), Fermat's and Wilson's Theorems, Euler's extension of Fermat's theorem(Only Statements) and its applications to find the remainder when divisible by a given number.

The topics in this module can befound in Chapter 3, sections A **and** B **and Chapter 9, sections** B **and** C **of Text[4].**

Textbooks

1. **Howard Anton, Irl C. Bivens, Stephen Davis,** *Calculus*, 10th Edition, John Wiley &Sons.
2. **B. S. Grewal**, *Higher Engineering Mathematics*, 42th Edition, Khanna Publishers.
3. **ZafarAhsan**, *Differential Equations and Their applications*,Second Edition,Prentice Hallof India.
4. **Lindsey N Childs** ,*A concrete Introduction to Higher Algebra*, Second Edition,Springer.

References

1. **Erwin Kreyszig**, *Advanced Engineering Mathematics*, 10th Edition,Wiley-India.
2. **S Barnard and J M Child**, *Higher Algebra*, Enlarged Edition, Macmillan And Company Limited.

REMARK

*Exercise and problems should be solved and graphed using a Computer Algebra System(CAS).

FOUNDATION COURSE I

AUCS121: COMPUTER FUNDAMENTALS AND ORGANIZATION

Credit: 3 **Instructional Hours: 4 per week**

1. AIM

- To create the overall generic awareness about the field of Information Technology and to impart knowledge in the functional organization of physical components and architecture of a computer.

2. OBJECTIVES

On the completion of this course, the student will be able:

- To get the basic concepts of Computers.
- To get the functional knowledge about PC hardware, operations and concepts.
- To understand the functional units of a standard PC and it's working.
- To understand the memory organization in a computer.

3. SYLLABUS

Module I: Characteristics of Computer; Von Neumann model; Inside a Computer: SMPS, Motherboard, BIOS, CMOS, Ports and Interfaces, Expansion Cards, Ribbon Cables, ASCII; Types of Input Devices, Types of Output Devices.

Module II: Memory Representation, Hierarchy, Memory Units: RAM (SRAM, DRAM); ROM; Secondary Storage Devices: Magnetic Tape, Magnetic Disk, Types of Magnetic Disks, Optical Disk, Types of Optical Disks; USB: Pen drive, External Hard Disk; Memory Stick; CPU Registers, Cache Memory, Operations in Cache memory, hit ratio; Virtual Memory.

Module III: Instruction Format; Instruction Cycle: Fetch Cycle, Execution Cycle; Instruction Set: CISC Architecture, RISC Architecture, Comparison; Memory Chips; Pipelining and Parallel Processing; Micro programmed Control and Hardwired Control.

Module IV: Input/Output Organization: Asynchronous Data Transfer, Programmed I/O (concepts only); Interrupts: Types of interrupts, processing interrupts, interrupt hardware and priority, DMA: DMA Controller, DMA Transfer Modes; I/O Processor.

4. REFERENCES

4.1 Core

- ***Introduction to Information Technology***, 2nd Edition, ITL Education Solutions Limited, Pearson.
- John D.Carpinelli, ***Computer systems Organization & Architecture***, Pearson Education.

4.2 Additional

- E.Balaguruswamy, ***Fundamentals of Computers***, McGraw hill, 2014
- Carl Hamacher, Vranesic, Zaky, ***Computer Organization*** 4th Edition, McGraw-Hill

4.3 Assignments and Activities:*Applications of Computers in various fields; Pioneers in IT; IT Policy, IT and Development; IT in India (major initiatives, key institutions, statistics), IT in Kerala (major initiatives, key institutions, statistics); Careers in IT; Computer faults: hardware & software; types of faults; diagnostic programs and tools; printer problems; monitor problems, problem diagnosis, organization of a modern PC.*

NB:- Activities and assignments are not meant for End Semester Examination

FOUNDATION COURSE II

AUCS122: DIGITAL ELECTRONICS

Credit : 3 **Instructional Hour: 3 per week**

1. AIM

- To impart basic knowledge in digital and logic circuits and to introduce basic concepts of data communications

2. OBJECTIVES

- To review basic electronic concepts
- To review data representation techniques
- To introduce student to basic concepts of digital logic
- To introduce the design of basic logical circuits.

3. SYLLABUS

Module I: Review of Basic Electronics:Diodes – half wave and full wave rectifiers. Zener diode, Zener diode as a voltage regulator, Bipolar Junction Transistors, Transistor configurations- CE characteristics, relation between transistor currents, Transistor biasing methods (CE configuration),

small signal CE amplifiers– graphical analysis of the small signal CE amplifier (frequency response, bandwidth and gain in dB) . RC phase shift oscillator, Astable multi-vibrator (555 timer).

Module II: Data Representation: Concept of number system bases – binary, decimal and hexadecimal number systems and conversion between each, Binary arithmetic: Addition, subtraction 1s and 2s complement system, multiplication. Codes: BCD, ASCII, Floating point representation

Module III: Boolean Algebra:Boolean functions: AND, OR and NOT, NAND, NOR and XOR. Truth tables, Combinational logic in Venn diagrams. Realization of other logic gates using NAND and NOR. Laws of Boolean Algebra, DeMorgan's theorems. Min terms, SOP expressions, Max terms, POS expressions, Karnaugh maps. Flip flops – SR flip flop, JK flip flop, Master Slave, D and T flip flops.

Module IV: Digital Circuits:Multiplexer, Full and half adders, Subtractors – half and full subtractors, Comparators – 1 bit and 2 bit, Counters, Decoder and display, shift registers, de-multiplexer and keyboard encoder.

4. REFERENCES

4.1 Core

- B L Theraja, ***Basic Electronics***, Chand Publications

4.2 Additional

- M Morris Mano, ***Digital Logic and Computer Design***, Pearson, 2013
- Thomas L Floyd, ***Digital Fundamentals***, Pearson, 2013

4.3 Assignments and Activities:*Miscellaneous Topics: Advances in Electronics: Evolution of Transistor Technology, Nano Technology, Molecular Electronics.*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE I

AUCS141: C PROGRAMMING

Credit: 3 **Instructional Hours: 3 per week**

1. AIM

- To Expose students to algorithmic thinking and problem solving and impart moderate skills in programming in an industry-standard programming language.

2. OBJECTIVES

- To expose students to algorithmic thinking and algorithmic representations.
- To introduce students to basic data types and control structures in C.
- To introduce students to structured programming concepts.
- To introduce students to standard library functions in C language.

3. SYLLABUS

Module I: Introduction to programming: Algorithm & Flow charts: Definitions, Symbols used to draw flowcharts, Program Writing – Structure of the Program, Source code, Object code, Executable

file, Variables and Constants, Rules for naming the Variables/Identifiers; Basic data types of C, int, char, float, double; storage capacity – range of all the data types;

Module II: Basic Elements: Operators and Expressions: Expression Evaluation (Precedence of Operators); simple I/O statements, Control structures, if, if else, switch-case, for, while, do-while, break, continue. Arrays: Defining simple arrays, Multi-dimensional arrays, declaration, initialization and processing.

Module III: Functions & Pointers: concept of modular programming, Library, User defined functions, declaration, definition & scope, recursion, Pointers: The & and * Operators, pointer declaration, assignment and arithmetic, visualizing pointers, call by value; call by reference, dynamic memory allocation. Storage classes.

Module IV: Advanced features: Array & pointer relationship, pointer to arrays, array of pointers. Strings: String handling functions; Structures and unions; File handling: text and binary files, file operations, Functions for file handling, Modes of files.

4. REFERENCES

4.1 Core

- Ashok N. Kamthene, ***Programming in C***, Pearson Education, Second edition

4.2 Additional

- E. Balaguruswamy, ***Programming in ANSI C***, McGrawhill, Sixth Edition

4.3 Assignments and Activities: *Pre-processor directives: #include, #define, macros with arguments, the operators# and ##, conditional compilations, multiple file programming; creating header files, program verification, algorithm efficiency analysis; int86 functions and graphic functions.*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE II

AUCS14P1: C PROGRAMMING LAB

Credit: 3 **Instructional Hours: 4 per week**

1. AIM

- To provide an opportunity for hands-on practice of algorithmic thinking and problem solving in an industry standard programming language.

2. OBJECTIVES

This course will provide hands-on practice in the following topics, under a variety of programming situations with a focus on writing, debugging and analyzing structured programs:

- basic data types in C
- basic control structures in C

- arrays, structures and files
- standard library functions in C language
- solving moderately complex problems involving the above and requiring selection of appropriate data structures and efficient algorithms

3. SYLLABUS

Part A

The C laboratory work will consist of 25-30 Experiments

1-15.Testing out and interpreting a variety of simple programs to demonstrate the syntax and use of the following features of the language: basic data types, operators and control structures.

Part B

16. 1-D Arrays: A variety of programs to declare, initialise, read, print and process 1-D arrays of various basic data types. Processing to include, selection, sum, counting, selective sum, selective counting, reversing etc.
17. Pointers: A large number of trivial programs involving all possible data types to familiarize the syntax of pointers in a variety of situations and to draw memory diagrams based on the observations.
18. Structures: A variety of programs to declare, initialise, read, print and process structures made up of a variety of data types and structures.
19. 2-D Arrays: A variety of programs to declare, initialise, read, print and process 2-D arrays of various basic data types. Processing to include, selection, sum, counting, selective sum, selective counting, reversing etc.
20. Array of Structures and Structure of Arrays: Programs to demonstrate declaration and processing of structure of arrays and array of structures.
21. Pointers to Arrays: A number of programs to demonstrate handling of 1-D and 2-D arrays using pointers and to draw memory diagrams based on the observations.
22. Pointers to Structures: A number of programs to demonstrate use of pointers to structures and to draw memory diagrams based on the observations.
23. Functions –I: Simple Examples of declaring and using functions of the following categories (i)no argument, no return, (ii) argument, no return, (iii) no argument, return, (iv) argument, return, all pass by value
24. Functions –II: Declaring and using functions with pass by reference, Passing and Returning structures, Recursive functions.
25. Files: Simple Example involving use of multiple files: declaring, opening, closing, reading from and writing to text files.
26. Files: Example involving use of multiple files: declaring, opening, closing, reading from and writing to binary files.
27. Library functions: A variety of Examples demonstrating (i) string processing functions (ii) a variety of selected library functions
28. Debugging programs involving syntactic and/or logical errors

29-30. Developing programming solutions to problems including program design, algorithmdevelopment and data structure selection.

4. REFERENCES

- Deitel&Deital, ***C: How to Program***, Pearson Education
- Alan R. Feuer, ***The C Puzzle Book***, Pearson Education
- YashvantKanetkar, ***Test Your C Skills***, BPB Publications, 3rd Edition

CORE COURSE III

AUCS14P1I: IT LAB

Credit: 3 **Instructional Hours: 4 per week**

1. AIM

- To make students undergo a hands-on experience in Latex and Spreadsheet software.

2. OBJECTIVES

By the end of the course, the students should be able to

- create attractive presentations in Latex
- generate worksheets in Open Office Calc and perform all basic computations

3. SYLLABUS

The IT Lab work will consist of 20-25 Experiments:

Part A– Latex

1. Building the File – Creating .tex file
2. Creating a Document class – A simple example to demonstrate the use of document class and to set the common font size, paper size and page formats in a .tex file
3. Simple typesetting – Examples to show the commands to add space, quotes, dashes, accents, special symbols, new line andalignment
4. Working with Fonts - Examples to demonstrate the commands to set different font sizes and type style
5. Page styles - Simple Examples to exhibit the use of \pagestyle{...} command, how the headers are set, to set the style of page numbers and formatting the lengths
6. Making Title and Abstract – Examples to demonstrate the commands to set the title part and the abstract of a document
7. Using Sectioning Commands – Various examples to show the use of commands to add chapter, sections, subsections, paragraph and subparagraph
8. Making Lists - Examples to exhibit the application of itemize environment, the enumerate environment and the description environment
9. Creating Tables - Various examples to show the use of tabular environment and formatting commands for the tabular environment and different packages to provide further facilities in forming a table
10. Using the verbatim environment - A simple example to demonstrate the utilization of the verbatim environment
11. Typesetting Mathematics – Simple examples to exhibit the capabilities of TeX in Mathematics typesetting through the commands to display the equations, superscripts and subscripts, roots and mathematical symbols, the use of custom commands, amsmath package, the various environments- equation*, multline*, split, gather*, align* and environments and commands for numbered equations, Matrices, dots, Delimiters, Fractions and various operators
12. Inserting figures - Various examples to show the use of figure environments and various commands used in the environment
13. Creating Table of Contents, Index, Glossary and Bibliographies - Examples to demonstrate the use of the commands to produce a table of content, formatting commands for table of contents,

commands to produce the index of a document, glossary ,thebibliography environment, citation commands, commands for multiple citations, index preparation author-year citation mode, citations without parenthesis, partial citations, citation aliasing

14. Using Boxes - A simple example to deliver the use of different boxes like LR boxes, paragraph boxes and nested boxes

Part B - Spreadsheet

15. Sheets: Examples to demonstrate working with Sheets – Inserting New Sheets, Moving and copying sheets, Deleting Sheets, Renaming Sheets, Sheet Protection, Inserting Page Break
16. Columns and Rows: Simple examples to demonstrate working with Columns and Rows – Inserting New Sheets, Moving and copying sheets, Deleting Sheets, Renaming SheetsInserting Rows and Columns, Deleting Rows and Columns, Grouping & Ungrouping Rows and Columns, Freezing Rows and Columns
17. Cells : A variety of examples to demonstrate working with cells – Merging, Splitting, Formatting Cell Borders, Formatting Cell backgrounds
18. Data: Examples using Data – Entering data, Font, Sorting data, Autoinput tool, Fill tool, Deleting data, Replacing data, Editing data, Paste Special, Formatting numbers, Formatting Fonts, Font Effects, Hiding and Showing Data; Conditional Formatting - Setting up conditional formatting, Defining conditions, Defining cell styles, Applying cell styles; Find & Replace - Find toolbar, Find & Replace dialog, Find & Replace options
19. Charts and Graphs : Simple Examples involving the use of Charts & Graphs - Insert Chart, Selecting Chart Types, Changing Chart Types, Changing chart graphic background
20. Functions and formulas: A variety of examples including the use of Formulas – Creating Formulas, Different Operators, Calculations linking sheets; Functions – Function Wizard, Nested Functions, Finding Mean, Median, Mode and Quartile, Subtotals
21. Filtering: A variety of examples to demonstrate the use of Filtering - Applying a standard filter, Applying an AutoFilter
22. Using Pivot Table: Example to show the use of Pivot Table – Creating Pivot Table, Options for Row and Column Fields

4. REFERENCES

- LibreOffice Documentation Team –***LibreOffice 5.1, Getting Started Guide***, 2016
- Indian TeX Users Group – ***LaTeX Tutorials – A Primer***, 2003
- Lamport, L. – ***LATEX: A Document Preparation System***, second edition, Addison-Wesley, 1994
- https://support.office.com/en-us/article/Excel-training
- https://www.sharelatex.com/learn

SEMESTER II

LANGUAGE COURSE II

AUEN211.5: ENGLISH FOR CAREER

Credit:2 **Instructional Hours: 3 per week**

Learning Objectives

- To introduce students to the language skills required for appearing in career oriented competitive examinations
- To frame modules of study that would develop the cognitive, logical, verbal and analytical skills necessary to succeed in competitive examinations.
- To provide the pattern of questions based on common models of competitive tests
- To provide sufficient practice in Vocabulary, Grammar, Comprehension and Remedial English from the perspective of career oriented tests.
- To help students to prepare for and appear in competitive examinations.

Learning Outcomes

The student will

- Acquire the necessary language skills required in the competitive job market.
- Acquire the cognitive, logical, analytical and verbal skills necessary to succeedin competitive examinations
- Become familiar with the pattern of questions usually asked in the competitive examinations
- Get sufficient practice in Vocabulary, Grammar, Comprehension and Remedial English
- Be able to prepare for and be successful in competitive examinations.

Course Description

Module I

Vocabulary

Ten passages with two sets of exercises- Passage based exercises (10) and general exercises(10) Identifying words from passages-meanings -synonyms-antonyms- one word substitutions-phrasal verbs- common errors in usage- common phrases and idioms-technical/professional/official usages- formal and informal registers in use – words to be used in sentences- confusing words- misspelt words. Key has been provided

Module III

Reading Comprehension

Comprehension of ten passages, with twelve questions each – Passages from different subject areas and different levels of complexity - Questions based on reading for information, understanding, learning, summarizing - Exercises to test comprehension, analytical and logical thinking, vocabulary skills and critical thinking.

ModuleIV

Remedial English

Set of hundred questions for correction of errors if/wherever necessary, with questions from all areas of Vocabulary, Usage and Grammar familiarized in the earlier sections. Key has been provided.

Note: The book is in the format of a workbook. Teachers can ask the students to write the answers in the spaces provided, or follow the directions given in the book.

Model Question Paper

Total Marks: 80 **Time: 3 Hours**

SECTION A

10 questions, based on errors in vocabulary and grammar (10 x 1 = 10 Marks)

SECTION B

Eight sets of questions, two in each set, from a total of 12. Fill in the blanks

(8 x 2=16Marks)

SECTION C

Six sets of questions, four in each set, from a total of nine. (6 x 4=24 Marks)

SECTION D

Two sets of questions, with fifteen (three sets of five each) in each set, from a total of our sets. (15 x 2=30 Marks)

Textbook Prescribed:

English for Success in Career: A Workbook

Edited by Dr.Bindu Nair, Associate Professor, Department of English, S.D College, Alappuzha, Dr.Sarita G., Assistant Professor, Department of English, Govt. Women's College, Thiruvananthapuram, R. Karthika, Assistant Professor, Department of English, S.D College, Alappuzha and M. Saritha, Assistant Professor, Department of English, S.D College, Alappuzha.

Publishers: Orient lackswan

Additional Reading:

Oxford English Language Reference. *Compact Oxford Dictionary, Thesaurus and Wordpower Guide*.OUP.

N.D.Turton and J.B. Heaton. *Dictionary of Common Errors*. Longman Ltd. 1998.
Jennifer Seidl and W. McMordie. *English Idioms and How to Use Them*. OUP 1978.

McCarthy, Michael and Felicity O' Dell. *English Vocabulary in Use*. Cambridge UP, 1994.

Roger Gower. *Grammar in Practice 1-6*. Cambridge UP. 2005, 2008.

Raymond Murphy. *Intermediate English Grammar*. Cambridge UP. 2005

Bridger, Nick and Alison Pohl. *Technical English: Vocabulary and Grammar.* Summertown Publishing, 2007.

Julie Moore. *Common Mistakes at Proficiency and How to Avoid Them*. Cambridge UP, 2005.
Rachel Roberts, *Practical English Grammar*. Viva Books, New Delhi.

V.K. Moothathu, *Concise English Grammar*. OUP 2013, 2014
F.T.Wood, *Remedial English Grammar*. Macmillan, 1979.

Michael A. Pyle and Mary Ellen Munoz. *Cliff's TOEFL Preparation Guide*. BPB Publications. 1992.

Bhatnagar, Mahesh et al. *General English – For All Competitive Exams.* Source Books, 2017.

Richa Dwivedi, *The Ultimate Guide to 21st Century Careers*. Hachette India, 2017.

Sangeeta Sharma, Gajendra Singh Chauhan. *Soft Skills: An Integrated Approach to Maximise Personality*. Wiley India. 2016.

Web sources:

https://www.educationforever.in

ttps://owl.purdue.edu/

https://www.pinterest.com/

www.naukri.com

COMPLEMENTARY COURSE II

AUMM231.2f - MATHEMATICS 1I:DISCRETE MATHEATICS

Credit: 3 **Instructional Hours: 4 per week**

MODULE1

MATHEMATICAL LOGIC **18 Hours**

Proposition and Connectives : Conditional and Biconditional Equivalence of Propositions, Tautology and Contradictions, Duality Theorem and its properties, Algebra of Proposition.

Normal Form: Principal Disjunctive, Principal Conjunctive Norma lForms and its applications using with and without truth tables

Theory of Inference: Rules of Inference-RuleP, RuleT and RuleCP, Consistent and Inconsistent premises, Indirect Method of Proof using these inference rules.

The topics in this module can be found in Chapter 1 of Text [1].

MODULE 2

PREDICATE LOGIC **18 Hours**

Quantifiers: Essential and Universal quantifier, Free and Bound Variables.

Rules of Specifications: Rule US, ES, UG, EG. Using these, convert a given statement into symbolic notation. Derivation from Premises using truth table and without using truth table.

The topics in this module can be found in the continuation of Chapter 1 of Text [1].

MODULE 3

SET THEORY **18 Hours**

Partition of Set: POSET - HASSE diagrams for partial ordering - lub, glb.

Lattices: Definition and Examples, principle of duality, Properties - Idem Potency, commutatively, associativity, absorption(sub lattices excluded).

GROUP THEORY: Definition, Examples, Order of a Group and its elements.

The topics in this module can be found in Chapter 2 and Chapter 5 of Text [1].

MODULE 4

CODING THEORYANDCOMBINATORICS **18 Hours**

Coding Theory: Group Code, Encoders and Decoders, Hamming Codes - Hamming distance, de coding and encoding function –correction and detection of errors in Group Codes-parity check matrix and its properties.

Combinatorics: Recurrencerelations of degree k with constant coefficients(Homogeneous and Non- homogeneous) and its solutions (Nonhomogeneous including Polynomial, Trigonometric sin(ax) or cos(ax) exponential –excluding their product combinations)-Generating function Method is also included.

The topics in this module can be found in Chapter 6 of Text [1].

Text

1. **T Veerarajan,** *Discrete Mathematics with Graph Theory and Combinatorics*, Tata McGraw- Hill, New Delhi, 2007.

References

1. **Seymour Lipschutz, Marc Lars Lipson** , *Discrete Mathematics*, Schaum's Solved Problems Series, McGraw-Hill InternationalEditions.
2. **Ralph P Grimaldi, B V Ramana**,*Discrete and Combinatorial Mathematics*, 5thEdition, PearsonEducation.

CORE COURSE IV

AUCS241: ENVIRONMENTAL STUDIES

Credit: 3 **Instructional Hours: 3 per week**

1. AIM

- To get awareness on natural systems and resources
- To get concepts of biodiversity and conservations
- To get basic knowledge on pollution and methods to solve these problems

2. OBJECTIVES

- To impart the knowledge on the environmental systems
- To impart the knowledge on the biodiversity and conservations
- To impart the knowledge on the environmental pollution and policies and practices
- To impart the knowledge on the impact of human communities on the environments

3. SYLLABUS

Module I: Environmental Studies – Introduction, Multidisciplinary nature, Scope and importance, Concept of sustainability and sustainable development. **Ecosystems –** Structure, function, Energy flow, food chains, food webs and ecological succession, Forest ecosystem, Grassland ecosystem, Desert ecosystem, Aquatic ecosystems, **Natural Resources** -Renewable and Non---renewable Resources, Land resources and use, land degradation, soil erosion and desertification, Deforestation - Causes and impacts due to mining, dam building on environment, forests, biodiversity and tribal populations, Water: Use and over---exploitation of surface and ground water, floods, droughts,

conflict over water - international & interstate, Energy resources- renewable and non-renewable, use of alternate energy sources, growing energy needs.

Module II: Biodiversity and Conservation - Levels of biological diversity: genetic, species and ecosystem diversity; Bio geographic zones of India; Biodiversity patterns and global biodiversity hot spots, endangered and endemic species of India, Threats - habitat loss, poaching of wildlife, man-wildlife conflicts, biological invasions; Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity, Eco-system and bio-diversity services: Ecological, economic, social, ethical, aesthetic and Informational value.

Module III: Environmental Pollution - Environmental pollution: types, causes, effects and controls; Air, water, soil and noise pollution, Nuclear hazards and human health risks, Solid waste management: Control measures of urban and industrial waste, Pollution case studies. **Environmental Policies & Practices -** Climate change, global warming, ozone layer depletion, acid rain and impacts on human communities and agriculture, Environment Laws: Environment Protection Act; Air (Prevention & Control of Pollution) Act; Water (Prevention and control of Pollution) Act; Wildlife Protection Act; Forest Conservation Act. Nature reserves, tribal populations and rights, and human wild life conflicts in Indian context.

Module IV: Human Communities and the Environment – Human population growth: Impacts on environment, human health and welfare., Resettlement and rehabilitation of project affected persons; case studies., Disaster management: floods, earthquake, cyclones and landslides., Environmental movements: Chipko Silent valley, Bishno is of Rajasthan., Environmental ethics: Role of Indian and other religions and cultures in environmental conservation., Environmental communication and public awareness, **Fire &safety:** Fire Detection Systems, Fire Control Systems, Care, maintenance, and Inspection

4. REFERENCES

4.1 Core

- "***Text book for Environmental Studies for undergraduate courses of all branches of Higher Education***", Erach Bharucha for University Grants Commission.
- ***Fire Safety Management Handbook***, Third Edition, Daniel E. Della-Giustina

4.2 Additional

- N Arumugan and V Kumaresan, "***Environmental Studies***", Saras Publication, 2014

4.3 Activities and Assignments: *C**ase studies of different environmental systems, pollution control organization report collections, different environmental protection activities and study reports.*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE V

AUCS242 : DATA STRUCTURES USING C

Credit: 3 **Instructional Hours: 4 per week**

1. AIM

- To introduce students to various data structures and their features and applicability.

2. OBJECTIVES

By the end of the course, students should:

- Be able to write well-structured programs in C
- Be familiar with data structures like array, structures, lists, stacks, queues, trees and graphs
- Able to appreciate various searching and sorting strategies

3. SYLLABUS

Module I: Sequential searching, binary searching, Hashing – linear hashing, hash functions, hash table searching, Sorting: bubble sort, selection sort, Stacks and Queues: FIFO and LIFO data structures – stacks using (i) pointers and (ii) arrays. Queues using (i) pointers and (ii) arrays, Operations on stack and queues; applications, polish notation.

Module II: Linked Lists: Concept of static versus dynamic data structures, implementation of linked lists using pointers, operations on linked lists: insertion, deletion and traversing. Doubly linked lists and circular linked lists, applications of linked lists.

Module III: Trees: Concept of linear versus non-linear data structures, various types of trees – binary, binary search trees. Creating a binary search tree, traversing a binary tree (in-order, pre-order and post-order), operations on a tree–insertion, deletion and processing, expression trees, implementation using pointers, applications.

Module IV: Graphs, graph traversal- depth-first and breadth-first traversal of graphs, applications.

4. REFERENCES

4.1 Core

- A.K.Sharma, ***Data Structures Using C***, Pearson, Second edition, 2011

4.2 Additional

- Nair A.S., Makhalekshmi,***Data Structures in C***, PHI, Third edition 2011.

4.3 Assignments and Activities:*Multi-way search trees, B-trees, Huffman trees, case studies.*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSEVI

AUCS243 : WEB PROGRAMMING

Credit: 3 **Instructional Hours: 3 per week**

1. AIM

- To Expose students to technology of web sites and to introduce various tools and languages required for technical and creative design of state-of-the-art web sites

2. OBJECTIVES

- To impart basic skills in moderately complex use of the following tools/scripts/languages: HTML, DHTML, CSS, Javascript.
- To impart necessary ability to choose the appropriate web tools/languages for creating state-of-the art websites
- To Expose students to current trends and styles in web design and applications

3. SYLLABUS

Module I: HTML - General Introduction to Internet and WWW; HTML: Structured language, Document types, Rules of html, Html tags, Head tags, Body tags, Headings , Divisions and Centering, Quotations, Preformatted text, Lists, Horizontal Rules, Block level elements, Text level elements, Character entities, Comments, Fonts, Tables: Table tags, Colors, Color names, Color values , Marquee tag.

Module II: Advanced HTML - Linking in html: Anchor tags, Layer tags, Link relationships, URL: Relative, Absolute, Image, Image maps, Frames: Layouts, Targeting, No frame tag, Floating frames, Audio, Embed tag, Forms: form tag and its attributes, Get, Post, Form field elements, Form accessibility enhancements: Access key, Tooltips, Browser-specific form accessibility improvements.

Module III: Javascript - Introduction to Javascript, Variables and data types, Declaring Variables, Operators, Control Structures, Conditional Statements, Loop Statements, Functions, Objects, Dialog Boxes, Alert Boxes, Confirm Boxes, Prompt Boxes, Javascript with HTML, Events, Arrays, Predefined objects, DHTML, Page Redirect, Void Keyword, Page Printing, String Methods, Error Handling, Validations, Publishing your Site, Cookies

Module IV: CSS: Style sheet Basics, Adding Style to a Document, CSS (**Cascading Style Sheet)** and HTML Elements, Selectors, Document Structure and Inheritance.

4. REFERENCES

4.1 Core

- V.K. Jain, ***Advanced Programming in Web Design***, Cyber Tech Publications

4.2 Additional

- Joel Sklar, ***Web Design Principles***, Vikas, 5thEdition
- ***The Complete Reference HTML & XHTML***, Thomas A Powell, 4thEdition
- H M Deitel, P J Deitel& A B Goldberg, ***Internet and Worldwide web programming: How toProgram***, 3/e, Pearson Education

4.3 Assignments and Activities: Introduction to **XML, Perl**: Basic control structures, Data types and Basic features

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE VII

AUCS24PI: DATA STRUCTURES USING C LAB

Credit: 3 **Instructional Hours: 4 per week**

1. AIM

- To develop skills in design and implementation of data structures and their applications.

2. OBJECTIVES

- To write and execute programs in C to solve problems using data structures such as arrays, linked lists, stacks, queues, trees, and search trees.
- To write and execute programs in C to implement various sorting and searching methods.

3. SYLLABUS

The laboratory work will consist of 15 – 20 experiments. The exercises may include the following:

Part A

1. Linked list: traversal, node deletion, node insertion in singly, doubly and circular lists
2. Implementation of different searching techniques
3. Implementation of different sorting techniques

Part B

4. Stacks: matrix representation and linked list representation: Push, Pop
5. Queues: matrix representation and linked list representation: Add, delete
6. Circular queue implementation
7. Evaluation of expression using stacks

Tree traversal

8. Evaluation of expression using binary trees.
9. Infix to postfix and prefix conversion
10. Creating and processing binary search tree

4. REFERENCES

- Seymour Lipschutz, ***Data Structures***, McGraw Hill Education

CORE COURSE VIII

AUCS24PII: WEB PROGRAMMING LAB

Credit : 3 **Instructional Hours: 4 per week**

1. AIM

- To Expose students to technology of web sites and to introduce various tools and languages required for technical and creative design of state-of-the-art web sites

2. OBJECTIVES

To impart basic skills in moderately complex use of the following tools/scripts/languages:

- HTML, DHTML, CGI Script, Perl, CSS, Java script.
- To impart necessary ability to choose the appropriate web tools/languages for creating state-of-the art web sites
- To Expose students to current trends and styles in web design and applications

3. SYLLABUS

The laboratory work will consist of 15-20 Experiments

Part A (HTML)

1. Practising basic HTML tags, text tags test styles, paragraph styles, headings, lists
2. Tables in HTML, Frames in HTML, nested frames, Link and Anchor Tags
3. Including graphics, video and sound in web pages, including Java applets
4. Layers & Image Maps
5. Creating animated Gifs
6. Cascading Style sheets
7. DHTML
8. HTML forms and Fields

Part B (JavaScript, Perl)

9. Exercises covering basic introduction to Perl
10. Exercises covering basic introduction to JavaScript
11. Development of a web site involving a variety of tools practiced above

4. REFERENCES

- Ivan Bayross, ***Web Enabled Commercial application Development Using… HTML, JavaScript, DHTML and PHP***, BPB Publications
- H M Deitel, P J Deitel& A B Goldberg, ***Internet and Worldwide web programming: How to Program***, 3/e, Pearson Education.

SEMESTER III

CORE COURSE IX

AUCS341: PROGRAMMING IN C++

Credit: 3 **Instructional Hours: 3 per week**

1. AIM

- To introduce the student to the basic concepts of object orientation and impart skills in an industry standard object oriented language

2. OBJECTIVES

On the completion of this course, the student will be able to

- understand the concepts of classes and object
- define classes for a given situation and instantiate objects for specific problem solving
- reuse available classes after modifications if possible
- possess skill in object oriented thought process

3. SYLLABUS

Module I: Concept of Object orientation – why related data and methods should be kept as a single unit – comparison with procedural and structured programming – Classes and objects – data abstraction, encapsulation, inheritance, polymorphism, dynamic binding, message passing. Advantages of object orientation – reusability, maintenance, security, comfort in programming. Input and output streams in C++, Basic data types and declarations

Module II: Classes and objects in C++, access modifiers, static members, friend functions, Constructors and Destructors, polymorphism, Operator Overloading and type conversion, anonymous objects

Module III: Inheritance- parent and child classes, private, public and protected inheritance, multiple inheritance and multi-level inheritance, Virtual base classes. C++ and memory models – new and delete operators, Heap, dynamic objects.

Module IV: Binding & Polymorphism: Early binding, Late Binding, Pointers to derived class objects, virtual functions, Pure virtual functions, abstract classes, object slicing, exception handling in C++: try, throw and catch

4. REFERENCES

4.1 Core

- Ashok N. Kamthane, ***Object oriented Programming with ANSI & Turbo C++***, Pearson

4.2 Additional

- H. M. Deitel and P. J. Deitel, ***C++: how to program***, Pearson Education
- Robert Lafore, ***Object Oriented Programming in Turbo C++***, Galgotia Publications

4.3 Activities and Assignments: *Working with files, Templates, Manipulating strings.*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE X

AUCS342: DATA COMMUNICATION &COMPUTER NETWORKS

Credit: 3 **Instructional Hours: 4 per week**

1. AIM

- To introduce the basics of data communications and to create an awareness of networking and different tools used

2. OBJECTIVES

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

- To introduce fundamental communication model.
- To introduce the concept of digital and analog signals
- Explain different components in each network layer
- Discuss different applications of it

3. SYLLABUS

Module I: Data Communications- Components, Data representation, Data flow**.**Introduction to Networks, Network criteria, Physical structures, Categories of Network, Interconnection of Networks, and Network models - OSI model-layers, TCP/IP protocol suite.Physical layer and Media - Data and signals-analog and digital,periodic analog signals, digital signals, bit rate, baud rate, bandwidth .Transmission impairments- attenuation distortion and noise.Multiplexing- FDM, TDM, WDM, Transmission Media-Guided and Unguided, Switching- circuit switched Network, packet-switched network, message switched network.

Module II: Data Link Layer-Error Detection and correction-types of Errors, Block coding, Hamming distance, Cyclic Codes, Checksum, Multiple Access Protocols - ALOHA, CSMA, CSMA/CD, FDMA, TDMA, CDMA, Ethernet-categories of Ethernet, Fast Ethernet.

Module III: Network Layer-Logical Addressing, IPV4 , IPV6, Network Layer- Forwarding, Unicast Routing Protocols- Distance vector routing, path vector routing, Multicast Routing Protocols(brief explanation only).

Module IV: Transport layer- Process to Process delivery- UDP, TCP, congestion control and Quality of Service- techniques to improve Quality of service. Application Layer- Domain Name System, FTP, TELNET, Electronic Mail, WWW, HTTP.

4. REFERENCES

4.1. Core

- Behrouz A. Forouzan, ***Data Communications and Networking***, McGraw-Hill Education (India) Pvt. Ltd.

4.2. Additional

- Andrew S. Tanenbaum, ***Computer Networks***, Pearson Education India

4.3 Assignments and Activities: *Ethernet network using hubs and switches, deploying a wireless LAN, Multimedia Support—SIP , Real-Time Transport Protocol (RTP), recent advances, high performance networks, next generation internet, asynchronous transfer mode.*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE XI

AUCS343: SYSTEM SOFTWARE

Credit: 3 **Instructional Hours: 3 per week**

1. AIM

- Provide an overall picture of the system related software

2. OBJECTIVES

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

- Explain the internal working of the system
- Discuss the principles of assemblers and narrate the working of loaders and linkers
- Discuss system development tools

3. SYLLABUS

Module I: Introduction &Assemblers: System software and machine architecture – The simplified Instructional Computer (SIC) - Machine architecture - Data and instruction formats – addressing modes - instruction sets - I/O and programming. Basic assembler functions - A simple SIC assembler – Assembler algorithm and data structures

Module II: Assemblers & Macro-processor: Machine dependent assembler features – Instruction formats and addressing modes – Program relocation - Machine independent assembler features -

Literals – Symbol-defining statements – Expressions - One pass assemblers and Multi pass assemblers. Basic macro processor functions - Macro Definition and Expansion – Macro Processor system software tools

Module III: Loaders And Linkers: Basic loader functions - Design of an Absolute Loader Machine dependent loader features - Relocation – Program Linking – Algorithm and Data Structures for Linking Loader - Machine-independent loader features – Automatic Library Search – Loader Options - Loader design options - Linkage Editors – Dynamic Linking – Bootstrap Loaders

Module IV: Compilers: Basic Compiler Functions: Grammars, Lexical Analysis, Syntactic Analysis, Code Generation. Machine Dependent Compiler Features – Intermediate Form of the program, Machine Dependent Code optimization. Machine Dependent Compiler features – Structured variables, machine-independent code optimization, Storage allocation. Compiler design options – Division into passes

4. REFERENCES

4.1 Core

- Leland L. Beck, *System Software – **An Introduction to Systems Programming***, 3rdEdition,Pearson Education Asia, 2006.

4.2 Additional

- D. M. Dhamdhere, ***Systems Programming and Operating Systems***, SecondRevised Edition, Tata McGraw-Hill, 2000.

4.3 Assignments and activities: *Case studies of lexical and syntax analyzers: LEX and YAAC, study on functions of various system software, study on compilation process of a program.*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE XII

AUCS344: VALUE EDUCATION

Credit: 2 **Instructional Hours: 3 per week**

1. AIM

- To get an awareness on different humanitarian and social tools like NSS, NCC
- Discuss the importance on the awareness on the situations like disaster management and organ donations

2. OBJECTIVES

- To impart the knowledge about the NSS, NCC
- To explore the idea on national integration and importance humanitarian values on national calamities like disaster management.
- To impart knowledge on the importance of organ donation and social welfares

3. SYLLABUS

Module I: NSS: *Introduction*: Basic Concepts, History, aims, Objectives -Emblem, Flag, motto, song, badge etc. Organizational structure, roles and responsibilities of various NSS functionalities, ***NSS Programmes and activities*** **-** regular activities, special camping Day camps, adaptation, Methodology of conducting survey, financial pattern of the scheme, other youth programme/schemes of GOI, Coordination with different agencies, Maintenance of the Diary. ***Youth Development Programmes*** **-**National Youth Policy, Youth development programmes at the National Level, State Level and voluntary sector, youth-focused and youth-led organisations. ***Additional Life Skills*****:** Positive thinking, self-confidence and self-esteem, setting life goals and working to achieve them, management of stress including Time management.

Module II: NCC: History, NCC –functions and duties, committees, aims, Moto, flag, Song, organization – RDC, CATC, NIC, Advanced leadership course, Army attachment camp, Hiking and trekking camps, Thal Sainik Camp, Vayu Sainik Camp, Nau Sainik camp, All Indian Yachting regatta, rock climbing camps, Naval Wing activities, Air Wing Activities, Youth exchange programme, achievements.

Module III: Disaster Management: Introduction to disasters – concepts, and definitions – disaster, hazard, vulnerability, resilience, risks - Bomb threat. Earth quake, Explosion, Hazardous material spill/release, Campus shooting, Terrorist incidence, Disaster classifications, causes, impacts – social, economic, political, environmental, health, psychosocial etc, Financial emergency, A sudden health emergency, Unexpected loss of income, Death in the family or other family emergency, Rent in arrears and risk of eviction, National disaster, Different impacts – in terms of caste, class, gender, age, location, disability, global trends in disasters, urban disasters, pandemics, complex emergencies, climate change. Disaster risk management – disaster relief – water, food, sanitation, shelter, health, waste management.

Module IV: Organ Donation: History, ethical issues in organ donation and transplantation, types of organ donation, How organ donation works, Legislation and global perspectives, Bioethical issues, Political issues, religious view point, distribution, suicide, controversies, public service announcements, how to become an organ donor, donation process, organ donation organization in different countries.

4. REFERENCES

- ***National Service Scheme – A Youth Volunteers Programme for Under Graduate students as per UGC guidelines***J.D.S.Panwar et al. Astral International. New Delhi.
- "***Army NCC cadet, Handbook specialized subjects***", Director General, 2013
- ***NCC, National Cadet Corps, A Concise Handbook of NCC Cadets for 'A', 'B' & 'C'Certificate Examinations"***
- ***Govt. of India Disaster Management act 2005***
- Govt. of India 2009, ***National Disaster Management Policy***

- Gupta Anil K, Sreeja S Nair, 2011, ***Environmental Knowledge for Disaster Risk Management***, NIDM, New Delhi
- ***Organ Donation: Medline Plus***
- Organdonor.gov, "***Timeline of Historical Events and Significant Milestones***", US Government Information on Organ Donation and Transplantation
- David Hamilton "***History of Organ Transplanation***", University of Pittsburgh Press 2012

4.3 Assignments and Activities: *Preparing various NSS activities reports, case studies, preparing program plans, preparation of disaster management activities and collection different agency reports, collection of different organ donation activity reports.*

NB: - Activities and assignments are not meant for End Semester Examination

CORE COURSE XIII

AUCS345: DATABASE MANAGEMENT SYSTEMS

Credit: 3 **Instructional Hours: 4 per week**

1. AIM

- To introduce basic concepts of data bases, and related techniques and tools

2. OBJECTIVES

- Be aware of basic concepts of data bases and data base management systems
- Be aware of concepts of relational data bases.
- Know to normalize relational data bases
- Skilled in using relational algebra and relational calculus
- Develop skills to write database queries

3. SYLLABUS

Module I: Introduction: evolution of data base systems, overview of database management systems, Relational data model, mathematical definition, candidate, primary and foreign keys, set operations on relations, insertion, deletion and update operations, attribute domains.

Module II: The E-R Model, Entities and attributes, 1-1 and many-1, many-many relationships. Security – Physical and Logical, Design and maintenance issues, integrity.

Module II: Relational algebra and relational calculus, Introduction to SQL, Table creation, selection, projection and join using SQL

Module IV: Functional Dependencies – Inference axioms, Normalization, 1NF, 2NF, 3NF and Boyce - Codd Normal forms, Lossless and lossy decompositions.

4. REFERENCES

4.1 Core

- Ramon A. Mata-toledo and Pauline K. Cushman, ***Fundamentals of Relational Data Bases***, SchaumOutlines, Tata McGraw Hill

4.2 Additional

- AtulKahate, ***Introduction to Data Base Management Systems***, Pearson Education

4.3 Assignments and activities:*Study of features of MS Access, Open Office Base, Oracle, mySQL, emerging areas.*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE XIV

AUCS34PI: PROGRAMMING IN C++ LAB

Credit: 3 **Instructional Hours: 4 per week**

1. AIM

- To provide an opportunity for hands-on practice of object oriented programming and problem solving in an industry-standard programming language.

2. OBJECTIVES

This course will provide hands-on practice in a the following topics, under a variety of programming situations with a focus on writing, debugging and analyzing object oriented programs:

- basic data types and control structures in C++
- managing classes and objects in a variety of situations
- solving moderately complex problems involving the above and requiring selection of appropriate structures and algorithms

3. SYLLABUS

The laboratory work will consist of 15-20 experiments, only by using class concept:

Part A

1. Testing out and interpreting a variety of simple programs to demonstrate the syntax and use of the following features of the language: basic data types, operators and control structures.
2. Solving a problem using (i) structures and (ii) classes and comparison between the two (the problem logic and details should be kept minimal and simple to enable focus on the contrast between the two methods, for example declaring result of a set of students defining the name and total marks in the program itself).
3. Class definitions and usage involving variety of constructors and destructors

Part B

4. Programs involving various kinds of inheritances
5. Programs involving operator overloading and type conversions
6. Programs involving virtual base classes, friend functions
7. Program to demonstrate early and late binding
8. Program to allocate memory dynamically
9. Program involving class and function templates
10. Programs to demonstrate(i) string processing (ii) file streams (iii) a variety of selected library functions
11. Exception handling
12. Handling of 2-D arrays using pointers
13. Debugging programs involving syntactic and/or logical errors

4.REFERENCES

- Deitel&Deitel, ***C++: How to Program***, Pearson Education

CORE COURSE XV

AUCS34PII: DBMS LAB

Credit: 3 **Instructional Hours: 4 per week**

1. AIM

This course will provide hands-on practice in the following topics, under a variety of computing situations with a focus on writing and analyzing SQL statements.

- Installing and configuring a proper SQL tool
- Database design and implementation
- Writing and analyzing SQL statements
- Create user interface and study the working of a data base in a front end application.

2. OBJECTIVES

The student should be made to

- learn to create and use a database
- be familiarized with a query language
- have hands on experience on DDL Commands
- have a good understanding of DML Commands and DCL commands
- familiarize advanced SQL queries
- be exposed to different applications

3. SYLLABUS

The laboratory work will consist of 15-20 Experiments. Experiments will cover creating tables including defining relations between them, practicing SQL, Experiments designed around a case study, miscellaneous topics including security, connecting databases to front-end applications. Some sample topics are given below:

Part A

1. SQL statement for creating, listing, dropping, checking, updating tables
2. Record manipulation using-insert, delete, update
3. Experiments that clarify the importance of keys (Except foreign key)
4. Queries with an Expression and a column alias
5. A simple query that aggregates (groups) over a whole table
6. A query with a literal string in the SELECT list
7. Queries with sub string comparison and ordering
8. Query using the "IS NULL" syntax to list (compare '=NULL' instead of IS NULL")
9. Finding values within a certain range
10. Using the --"BETWEEN" keyword
11. SQL functions (String, Numeric, Date functions)
12. Aggregate Functions

Part B

13. A Join between two tables (Natural Join, Theta Join etc.)
14. Foreign Key
15. Nested queries
16. The EXISTS and UNIQUE function in SQL
17. Renaming attributes and joined tables
18. Statements related with VIEWs

4. REFERENCES

- Elmasri&Navathe,***Fundamentals of Database Systems***, Pearson Education.
- Ramon A. Mata-toledo and Pauline K. Cushman, ***Fundamentals of Relational Data Bases***, Schaum Outlines, Tata McGraw Hill
- Abraham Silberschatz, Henry F. Korth, ***Database System Concepts***, McGraw Hill

SEMESTER IV

CORE COURSE XVI

AUCS441: OPERATING SYSTEMS

Credit: 3 **Instructional Hours: 4 per week**

1. **AIM** To introduce students to basic functions and the theoretical underpinnings of modern operating systems

2. OBJECTIVES

To introduce students to:

- Fundamental concepts of systems software and functions of operating systems as a resource manager
- Strategies for constrained resource allocation and process scheduling
- Memory and I/O Management techniques
- Salient features of popular operating systems.

3. SYLLABUS

Module I: Introduction to operating system: Introduction, Operating system structures-Operating System Operations, operating system services, user operating system interface, system programs, system calls, Types of System Calls, operating system structure. **Process Management**: Process concept, Process Scheduling, Operations on processes, Inter-process communication, Threads-Overview, Multithreading model, Thread Libraries, Threading issues; CPU Scheduling: Basic concepts, scheduling criteria, Scheduling algorithms.

Module II: Process synchronization: Background, Critical section problem, Peterson's solution, Semaphore, Classical synchronization problem-bounded buffer problem, reader/writer problem. The Dining Philosophers problem. **Deadlocks:** deadlock characterization, methods for handling deadlock-deadlock prevention, deadlock avoidance, deadlock detection, deadlock recovery.

Module III: Memory Management & Protection: Basic Hardware, Address binding, Logical versus physical address space, Swapping, Contiguous memory allocation- memory mapping and protection, memory allocation, fragmentation. Non-contiguous allocation – paging, segmentation. **Virtual memory**- Demand Paging, page replacement, Allocation of Frames, Thrashing, Allocating Kernel Memory. Protection and Security: Protection -principles of protection, domain of protection, access matrix, access control; Security- threats, user authentication.

Module IV: Storage management: File system Interface - file concept, access methods, directory structure, File Sharing, **File system implementation**- file system structure &implementation, directory implementation, allocation methods, free space management; Mass storage management -

disk structure, disk scheduling, RAID;I/O Systems – I/O hardware, Application I/O interface, kernel I/O subsystem.

4. REFERENCES

4.1 Core

- Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, ***Operating System Concepts***7th edition

4.2 Additional

- Achyut S Godbole, ***Operating systems***, McGRawhill, Third Edition

4.3 Assignments and Activities:*Case study of popular Operating Systems like Android, Windows, Sun Solaris, IOS etc*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE XVII

AUCS442: SOFTWARE ENGINEERING

Credit: 3 **Instructional Hours: 4 per week**

1. AIM

- To enable the students to have a thorough understanding of the activities in development projects using (a) Structured Analysis and Design and (b) Object Oriented Analysis and Design

2. OBJECTIVES

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

- appreciate the importance of having a process for software development
- understand the various activities undertaken for a software development project following the Function oriented Design & Object oriented design
- understand the issues in code design and development
- test software developed using SSAD and OOAD methodologies
- have in depth knowledge about the different OOAD Themes and compare them with SSAD

3. SYLLABUS

Module I: Introduction to software engineering- scope of software engineering. Software engineering a layered technology – processes, methods and tools. Capability maturity model (CMM), ISO 9000. Software Development Life Cycle Model: Waterfall Model, Prototyping, iterative development, Spiral Model, time-boxing model, Agile model; Comparison of different Life Cycle Models.

Module II: Software Requirements, Overview of SA/SD Methodology, Requirements Specification: Need for SRS, Characteristics of an SRS, Components of an SRS, Specification Languages, Structure of a Requirements document. **Planning phase** – project planning objective,

software scope, empirical estimation models- COCOMO, single variable model, staffing and personal planning.

Module III: Design phase-Function oriented Design – design process, principles, concepts, effective modular design, top down, bottom up strategies, coupling, cohesion, design notations-structure charts, structured design, Data Flow Diagrams, Developing the DFD Model of a system, Entity Relationship Diagram, Developing ERD of a system, Decision Trees, Decision Tables, Structured English. **Object Oriented Design-** Object Oriented Design using UML, Class Diagram, Use Case Diagram, Sequence Diagram, Collaboration Diagram.

Module IV: Coding and testing: common coding errors, structured programming, codingstandards, incremental coding process, test driven development, source code control andbuild, refactoring, verification- code inspections, static analysis, unit testing, combiningdifferent techniques. Testing-error, fault and failure, test oracles, test cases, Black BoxTesting, Equivalence Class Partitioning, Boundary Value Analysis, Cause Effect Graphing,White Box Testing- control flow based and data-flow based testing, test plan, test casespecifications, defect logging and tracking, Comparison of Different Techniques. Maintenance-Overview of maintenance process, types of maintenance, Reverse Engineering.

4. REFERENCES

4.1 Core

- Roger S. Pressman, ***Software Engineering*****,** McGraw Hill

4.2 Additional References

- Rajib Mall, ***Fundamentals of Software Engineering***, Second Edition, PHI
- Ian Sommervile, University of Lancastor, ***Software Engineering***, Pearson Education
- K. K. Aggarwal and Yogesh Singh, ***Software Engineering***, New age International Publishers.
- PankajJalote, ***An Integrated Approach to Software Engineering***, Narosa Publishing House
- Waman S. Jawadekar, ***Software Engineering***, McGraw hill, 2013

4.3 Journals and Magazines

- Software Development, CMP Media.
- Software Quality Professional, ASQ

4.4 Activities and Assignments: *Preparing various documents, case studies, preparing test plans, UML diagrams, Metrics for various development phases, Agile Programming Methodologies, extreme Programming, Formal Methods, CASE Tools.*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE XVIII

AUCS443: INTERNET OF THINGS

Credit: 3 **Instructional Hours: 3per week**

1. AIM

- To introduce the basic concepts of the convergence of operational technology (OT) and informational technology (IT)

2. OBJECTIVES

- To get a deep dive into IoT network engineering, from smart objects and the network that connects them to applications, data analytics, and security.
- To guide through the different types of smart objects, from those that simply record information to those that are programmed to perform actions in response to changes.
- To guide through the different common application protocols to generic and web-based protocols.
- To get basic knowledge about the security practices for IT and OT and details how security is applied to an IoT environment.

3. SYLLABUS

Module I: Introduction to IoT: - Genesis of IoT, Digitization, Impact, Connected Roadways - Challenges- safety, mobility, environment, Connected Factory -industry – mechanical assistance, mass production, electronics and control, integration, Smart Connected Buildings – heating, ventilation, HVAC systems, BAS System, BACNet, Smart Creatures, Convergence of IT and OT, IoT Challenges – Scale, Security, Privacy, Big data and data analytics. **IoT Network Architecture and Design: -** Drivers Behind New Network Architectures, Comparing IoT Architectures, A Simplified IoT Architecture, The Core IoT Functional Stack, IoT Data Management and Compute Stack

Module II: Engineering IoT Networks: Smart Objects **-** Sensors, Actuators, and Smart Objects, Sensor Networks, **Connecting Smart Objects:** Communications Criteria IoT Access Technologies

Module III: IP as the IoT Network Layer: Business Case for IP, Need for Optimization, Optimizing IP for IoT, Profiles and Compliances, **Application Protocols for IoT:**Transport Layer, IoT Application Transport Methods

Module IV: Securing IoT: A Brief History of OT Security, Common Challenges in OT Security, How IT and OT Security Practices and Systems Vary, Formal Risk Analysis Structures: OCTAVE and FAIR, The Phased Application of Security in an Operational Environment, **Public Safety:** Overview of Public Safety, An IoT Blueprint for Public Safety, Emergency Response IoT Architecture, IoT Public Safety Information Processing, School Bus Safety.

4. REFERENCES

4.1 Core

- David Hanes, " IoT***Fundamentals: Networking Technologies, Protocols and Use Cases for the Internet of Things***", Cisco Press, Pearson, 2017.

4.2 Additional

- Graham meikle, "***The internet of things***", polity press, 2017
- Andrew Minteer, "***Analytics for the internet of things: Intelligent analytics for your intelligent devices***", Packt publishing, 2017
- Jan Holler, VlasiosTsiatsis, Catherine Mulligan, Stefan Avesand, StamatisKarnouskos, David Boyle, **"*From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence*",** 1st Edition, Academic Press, 2014.
- Peter Waher**, "*Learning Internet of Things***", PACKT publishing, BIRMINGHAM – MUMBAI
- Bernd Scholz-Reiter, Florian Michahelles, "***Architecting the Internet of Things***", ISBN 978-3-642-19156-5 e-ISBN 978-3-642-19157-2, Springer

4.3 Activities and Assignments: *Wireless sensor networks*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE XIX

AUCS444: PHP

Credit: 3 **Instructional Hours: 3 per week**

1. AIM

- To expose students to technology of web sites and to introduce various tools and languages required for technical and creative design of state-of-the-art web sites

2. OBJECTIVES

- To impart basic skills in moderately complex use of the following tools/ scripts/ languages:
- To choose the appropriate web tools/languages for creating state-of-the art web sites
- To expose students to current trends and styles in web design and applications

3. SYLLABUS

Module I: Overview of PHP, Benefits and drawbacks in running PHP as a Sever Side Script, PHP Language Basics: The building blocks of PHP: variables, globals& super globals Data types: Set type, type casting, test type, Operators & Expressions, Flow control functions in PHP, Functions: Defining a function variable scope, calling a function returning values, setting default values for arguments, passing variable reference, built in functions

Module II: Arrays: Creating arrays (associative & multidimensional), Array related functions. Working with Objects, Working with string functions: Formatting strings, Using Date and Time functions. Forms in PHP: Form elements, adding elements to a form, creating a simple input form, combining HTML & PHP code on a single page, redirecting the user, creating a send mail form, File upload form, working with files and directories.

Module III: Cookies: Introduction, different types of cookies, setting a cookie with PHP, deleting a cookie, session function overview: starting a session, working with session variables, passing session IDs in the query string, destroying sessions &unsetting variables, Working with images.

Module IV: Database concepts: Open source database software: MySQL features MySQL data types: Numeric, date & time, string Table creation in MySQL: insert, select, where clause, ordering the result, like operator Selecting Multiple tables: using join, using queries Modifying records: update command, replace command, delete command date & time functions in MySQL Interacting with MySQL using PHP: connecting to MYSQL ,Executing queries, Retrieving error messages, inserting data with PHP, retrieving data with PHP.

4. REFERENCES

4.1 Core

- Julie C.Meloni, ***PHP, MySQL and Apache***,Pearson Education

4.2 Additional

- Ivan Byross, ***PHP for Beginners***

4.3 Assignments and activities:*Sample programs which connects PHP and database, case studies.*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE XX

AUCS445: PROGRAMMING IN JAVA

Credit: 3 **Instructional Hours: 3 per week**

1. AIM

- To introduce students to basic features of Java language and selected APIs

2. OBJECTIVES

- Let students install and work with JDK, also make them aware the use of java doc.
- Practice basic data types, operators and control structures in Java
- Practice basic handling of classes and objects in Java
- Introduce the following selected APIs: I/O, Strings, Threads, AWT, Applet, Networking
- Idea to approach and use a new package

3. SYLLABUS

Module I: A simple Java Application, a simple Java Applet , Brief History of Java, Special Features of Java, Data Type & Operators in Java, Arrays, Objects, the Assignment Statement, Arithmetic Operators, Relational and Logical Operators in Java, control Structures, The Java Class, Constructor, Finalizers, Classes inside classes: composition

Module II: Inheritance & Interface, Deriving Classes, Method Over-riding, Method Overloading, Access Modifiers, Abstract Class and Method, Interfaces, Packages, Imports and Class Path.

Module III: Exception Handling, The Try-Catch Statement, Catching more than one Exception, The Finally Clause, Generating Exceptions, Threads: Introduction, Creating Threads in Applications, Method in Thread Class, Threads in Applets.

Module IV: Java APIs – overview of APIs, IO Packages, Java Input Stream Classes, Java Output Stream Classes, File Class, Graphic & Sound: AWT and Swing, Graphic methods, Fonts, Loading and Viewing Images, Loading and Playing Sound, AWT & Event Handling, Layouts, **JDBC.**

4. REFERENCES

4.1 Core

- ***Java Programming***, *Schaum Outline Series*

4.2 Additional

- Deitel, ***Java: How To Program***, Pearson Education
- E Balagurusamy, "***Programming with Java – A Primer***", McGraw Hill, 2017

4.3 Assignments and Activities:*Creation of simple programs with interfaces, concepts of OOPs.*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE XXI

AUCS44PI: PROGRAMMING IN JAVA LAB

Credit: 3 **Instructional Hours: 4 per week**

1. AIM

- To provide an opportunity for hands-on practice in Java.

2. OBJECTIVES

This course will provide hands-on practice, under a variety of programming situations with a focus on writing, debugging and analysing object oriented programs. The course shall include the following

- basic data types and control structures in Java
- installing and using JDK
- writing applications and applets
- managing classes and objects in a variety of situations
- using i/o, string, and threads
- solving moderately complex problems involving the above

3. SYLLABUS

The laboratory work will consist of 15-20 Experiments.

Part A

1. Testing out and interpreting a variety of simple programs to demonstrate the syntax and use of the following features of the language: basic data types, operators and control structures.
2. Class definitions and usage involving variety of constructors and finalizers
3. Programs involving various kinds of inheritances,
4. Program involving Method Over-riding, Method Over-loading
5. Program involving Abstract Class and Methods

Part B

6. Program involving Interface,
7. Program to demonstrate creation and handling of packages, their imports and Class Path.
8. Programs involving a variety of Exception Handling situations
9. Program to define a class that generates Exceptions and using objects of the class.
10. Program involving creating and handling threads in applications and applets.

11-12. Programs to demonstrate methods of various i/o classes

13. Programs to demonstrate methods of string class
14. Program to demonstrate AWT/Swing graphic methods
15. Program for Loading and Viewing Images, Loading and Playing Sound
16. Programs to demonstrate various Layouts

17-18. Programs to demonstrate event handling

19. Program to demonstrate simple server-client (using a single m/c both as client and server)
20. Debugging programs involving syntactic and/or logical errors

4. REFERENCES

- ***Java Programming***, *Schaum Outline Series*
- Deitel, ***Java: How To Program***, Pearson Education
- E Balagurusamy, "***Programming with Java – A Primer***", McGraw Hill, 2017

CORE COURSE XXII

AUCS44PII: PHP LAB

Credit: 3 **Instructional Hours: 4 per week**

1. AIM

- To earns skill set to develop online applications using the open sources PHP and MySQL.

2. OBJECTIVES

This course will provide hands-on practice in the following topics:

- basic data types and control structures
- Arrays, Cookies and Sessions
- Connectivity between PHP and MySQL and develop programs to add records, retrieve records and delete records from a table.

3. SYLLABUS

Setup WAMP/XAMPP Server or Setup Apache, MySQL and PHP separately in your PHP Lab.

Part A

1. Simple PHP program that displays a welcome message.
2. Write a PHP program to generate a random number between 1 and 100.
3. Modify above program to accept range of the random number from HTML interface.
4. Programs involving various control structures like if, else, elseif/else if', Alternative Syntax for 'if, else, elseif/else if'
5. Programs involving various control structures like while, do-while, for, foreach, switch, break, continue. Try alternative syntax for while, do-while, for, foreach, switch.
6. Programs involving the declare, return, require, include, require- once, include once and goto.
7. Programs to demonstrate PHP Array functions, PHP Array Sorting, PHP Key Sorting, PHP Value Sorting, PHP Multi Array Sorting, PHP Array Random Sorting,
8. Programs to demonstrate PHP Array functions. PHP Array Reverse Sorting, Array to String Conversion, Implode() function, String to Array, Array Count, Remove Duplicate Values
9. Programs to demonstrate PHP Array functions. array Search, Array Replace, Array Replace Recursive, Array Sub String Search
10. Demonstrate the use of regular expression to compare two strings.
11. Extract Domain name from URL
12. Find the number of rows from a mysql database for your query.
13. Generate a Guestbook which will allow your website visitor to enter some simple data about your website.
14. Develop a PHP program for Email Registration.
15. Develop a project for making Application form and performing Degree Admission On-line.

Part B[Sample Questions]

16. Write a PHP script to find the factorial of a given number.
17. Write a PHP script to find the sum of digits of a given number.
18. Write a PHP script to find whether the given number is a prime or not.
19. Write a PHP script to demonstrate the use of break, continue statements using nested loops.
20. Write a PHP script to display the Fibonacci sequence with HTML page.
21. Write a PHP script to create a chess board.
22. Write a PHP script using built-in string function like strstr(), strops(), substr_count(), etc
23. Write a PHP script to transform a string to uppercase, lowercase letters, make a string's first character uppercase.
24. Write a PHP script that inserts a new item in an array in any position.
25. Write a PHP function to check whether all array values are strings or not.
26. Write a PHP script to count number of elements in an array and display a range of array elements.
27. Write a PHP script to sort a multi-dimensional array set by a specific key.
28. Write a PHP script using a function to display the entered string in reverse.
29. Write a PHP script using function for sorting words in a block of text by length.
30. Write a PHP script for creating the Fibonacci sequence with recursive function.
31. Write a PHP script using pass by value and pass by reference mechanisms in passing arguments to functions.
32. Write a PHP script to demonstrate the defining and using object properties.
33. Write a PHP script to demonstrate the inheritance.
34. Write a PHP script to demonstrate the object overloading with _get(), _set(), and _call().
35. Write a PHP script to demonstrate the overloading property accesses with _get() and _set().
36. Write a PHP script to demonstrate the object overloading with _get(), _set(), and _call().
37. Write a PHP script to demonstrate the overloading property accesses with _get() and _set().
38. Write a PHP script to demonstrate the use interfaces.
39. Write a PHP script using constructors and destructors.
40. Write a PHP application to handling HTML forms with PHP script.
41. Write a PHP script to create a file, write data into file and display the file's data.
42. Write a PHP script to check and change file permissions, copying, renaming and deleting files.
43. Write a PHP application for connecting to MySQL and reading data from database table.
44. Write a PHP application for inserting, updating, deleting records in the database table.
45. Write a PHP application for student registration form.

4. REFERENCES

- Julie C.Meloni, ***PHP, MySQL and Apache***,Pearson Education
- Ivan Byross, ***PHP for Beginner***

SEMESTER V

CORE COURSE XXIII

AUCS541: COMPUTER GRAPHICS

Credit : 3 **Instructional Hours: 3per week**

1. AIM

- To introduce theoretical concepts behind computer graphics
- To introduce the algorithms, tools and techniques for implementing the same.

2. OBJECTIVES

On completion of this course, students should be able to:

- handle basic graphic primitives in C/C++ for developing 2D and 3D graphics
- program basic scan-conversion algorithms
- apply various transformations to 2D and 3D graphic objects
- derive various projections of 3D objects
- give realistic rendering to 3D wireframe objects
- be familiar with current trends in computer graphics

3. SYLLABUS

Module I: Introduction: graphic data representation, concept of pixels, resolution, aspect ratio, Raster scan display, Random Scan display, video adapter, frame buffer, display technology-CRT, LCD, LED, smart devices (feature wise comparison only), Output Primitives: Straight Line, DDA algorithm, Bresenham's Line Algorithm, Circle- Mid Point Circle Algorithm, polygon filling algorithms- boundary fill, scan-line algorithm, Aliasing and Anti-aliasing.

Module II: Two dimensional Transformations: Translation, scaling, fixed point scaling, rotation, reflection, transformation with respect to arbitrary points. Application of homogeneous coordinates for uniform matrix operations, composite transformations, Windowing and clipping: Window to viewport transformation, Clipping- Point clipping, Line Clipping, Cohen-Sutherland Line Clipping algorithms, Polygon Clipping-Sutherland-hodge man algorithm.

Module III: 3D Concepts and Techniques: 3D display techniques, 3D Transformations, 3D modelling schemes, Projection-parallel projections, perspective projection, Visible Surface Detection- Hidden Surface removal Algorithms-scan line method, Z-buffer method.

Module IV: Colour Illumination methods: color models-RGB, HSI, CMYK, Illumination model and light sources, Specular reflection, Intensity attenuation, shadow, Polygon Shading methods, animation**,** morphing-tweening, warping (Concepts only) zooming, panning, rubber band lines (concepts only)

4. REFERENCES

4.1 Core

- Amarendra N Sinha and Arun D Udai, ***Computer Graphics***, McGraw Hill publications

4.2 Additional

- Donald Hearn, M. Pauline Baker, ***Computer Graphics (C Version) 2/e,*** Pearson

4.3 Assignments and activities: *Studies on various algorithms, creation of sample programs for implementing graphics algorithms, programs to implement the transformation algorithms. Case studies on new trends in graphics and image processing tools.*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE XXIV

AUCS542: DESIGN & ANALYSIS OF ALGORITHMS

Credit: 3 **Instructional Hours: 4 per week**

1. AIM

- To make students able to devise and analyze new algorithms by themselves.

2. OBJECTIVES

On completion this course, student should:

- Be able to analyze the complexity of algorithms
- Be able to select good algorithms from among multiple solutions for a problem
- Have better knowledge on fundamental strategies of algorithm design and awareness on complex algorithm design strategies
- Implement some typical algorithms

3. SYLLABUS

Module I: Algorithm Analysis: Algorithm, Properties of a good algorithm, efficiency considerations, Complexity: time complexity, space complexity, Asymptotic notations: Big O notation, best case, worst case, average case, simple examples, recursion and its elimination- recursive and no-recursive algorithms for binary search.

Module II: Algorithm design techniques-Divide and conquer method: binary search as a divide-and-conquer algorithm, finding maximum and minimum, Strassen's matrix multiplication, Greedy method: Knapsack problem, minimum cost spanning trees, Prim's algorithm, Kruskal's algorithm.

Module III: Dynamic programming: principle of optimality, all pair shortest paths, single source shortest paths, travelling sales person's problem, Back tracking: implicit constraints and explicit constraints, 8 queen's problem, Branch and bound: LC search

Module IV: Standard Algorithms: sorting- quick sort, merge sort, complexity of sorting algorithms, Deterministic and non-deterministic algorithms, NP-hard and NP complete- basic concepts.

4. REFERENCES

4.1 Core

- AnanyLevitin, ***Introduction to design and analysis of algorithms***, Pearson, Second Edition

4.2 Additional

- Ellis Horowitz, SartajSahni, SanguthevarRajasekharan –***Computer Algorithms / C++***, SecondEdition- Universities Press.

4.3 Assignments and Activities:*Studies on complexities of various algorithms, best case, average case worst case analysis.*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE XXV

AUCS543: PYTHON PROGRAMMING

Credit: 3 **Instructional Hours: 3per week**

1. AIM

- To introduce the student to the basic features of python programming and impart skills in an Industry standard programming language

2. OBJECTIVES

On the completion of this course, the student will be able to

- Understand the concepts of python programming
- Create new GUI based programming to solve industry standard problems

3. SYLLABUS

Module I: Introduction to Python - Features of Python - Identifiers - Reserved Keywords - Variables Comments in Python – Input , Output and Import Functions - Operators – Data Types and Operations – int, float, complex, Strings, List, Tuple, Set,Dictionary - Mutable and Immutable Objects – Data Type Conversion - Illustrative programs: selection sort,insertion sort, bubble sort

Module II: Decision Making -conditional (if), alternative (if-else), if..elif..else -nested if - Loops for,range() while, break, continue, pass; Functions: return values, parameters, local and global scope, function composition, recursion; Strings: string slices, immutability, string functions and methods,

string module; Lists as arrays. Illustrative programs: square root, gcd, exponentiation, sum of an array of numbers, linear search, binary search, bubble sort, insertion sort, selection sort

Module III: Built-in Modules - Creating Modules - Import statement - Locating modules - Namespaces and Scope - The dir() function - The reload function - Packages in PythonFiles and exception: text files, reading and writing files Renaming and Deleting files Exception handling exceptions, Exception with arguments, Raising an Exception - User defined Exceptions - Assertions in

Module IV: GUI Programming- Introduction – Tkinter Widgets – Label – Message Widget – Entry Widget – Text Widget – tk Message Box – Button Widget – Radio Button- Check Button – Listbox- Frames _ Toplevel Widgets – Menu Widget

4. REFERENCES

4.1 Core

- "Taming PYTHON By Programming", Jeeva Jose Khanna Publications

4.2 Additional

- Kenneth A. Lambert, "***Fundamentals of Python: First Programs***", CENGAGE Learning,2012.
- Paul Gries, Jennifer Campbell and Jason Montojo, "***Practical Programming: An Introductionto Computer Science using Python 3***", Second edition, Pragmatic Programmers,LLC,2013.
- Timothy A. Budd, "***Exploring Python***", Mc-Graw Hill Education (India) Private Ltd.

4.3 Assignments and activities:*case studies on GUI based program developments, free software foundation data collections, new graphic standards programs generations.*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE XXVI

AUCS54PI: PYTHON PROGRAMMING LAB

Credit: 3 **Instructional Hours: 4 per week**

1. AIM

- To provide an opportunity for hands-on practice in Python Programming.

2. OBJECTIVES

This course will provide hands-on practice in the following topics, under a variety of programming situations:

- basic data types and control structures in Python

- use functions in Python
- Represent compound data
- File handling, Exception handling & GUI programming

3. SYLLABUS

The laboratory work will consist of 10-15 Experiments.

Part A

1. To write, test, and debug simple Python programs.
2. To implement Python programs with conditionals and loops.
3. Use functions for structuring Python programs.
4. Represent compound data using Python lists, tuples, dictionaries.

Part B

5. Read and write data from/to files in Python.
6. Programs to demonstrate creating and handling of modules and packages
7. Programs involving a variety of Exception Handling situations
8. GUI programming

4. REFERENCES

- "***Taming PYTHON By Programming***", Jeeva Jose, Khanna Publications
- Kenneth A. Lambert, "***Fundamentals of Python: First Programs***", CENGAGE Learning,2012.
- Paul Gries, Jennifer Campbell and Jason Montojo, "***Practical Programming: An Introductionto Computer Science using Python 3***", Second edition, Pragmatic Programmers,LLC,2013.
- Timothy A. Budd, "***Exploring Python***", Mc-Graw Hill Education (India) Private Ltd.

CORE COURSE XXVII

AUCS544: MAJOR PROJECT PHASE I

Credit: 2 **Instructional Hours: 4 per week**

1. AIM:

- To provide an opportunity to produce and develop socially useful software.

2. OBJECTIVE

- To provide an opportunity for structured team work and project management.
- To provide an opportunity to practice the various phases in the Software Development Life Cycle
- To introduce the prospect of effective technical documentation and presentation.
- To provide an opportunity to practice time, resource and person management

3. GUIDELINES FOR MAJOR PROJECT (PHASE I)

The following guidelines should be adhered to:

- The guide lines given for the course AUCS 644has to be followed for this course.
- An analysis and design report should be submitted at the end of this course.
- Report should be typeset in Latex.
- Up to design should be finished in 5th semester.
- Team size should preferably be three with a maximum limit of 4 members.
- Individual projects may be permitted in exceptional cases, for valid reasons
- Internal guide(s) should be assigned to each team.
- Two interim reports (one after analysis and another after design) should be submitted to internal guides.
- The number of records to be submitted is limited to team size + one (Departmental copy). Hard binding of reports is optional.
- The report format guidelines used to document Major Project should be followed for making the final report and evaluation will be made on the same grounds.

4. EVALUATION

4.1 Criteria for external evaluation of Major Project (Phase I)

External evaluation is done by an external examiner appointed by the College. The following components are to be assessed for the End Semester Evaluation of the **Major Project (Phase I):**

- ✓ Documentation- 30 marks
- ✓ Presentation of work- 25 marks
- ✓ Viva - 25 marks

Total - 80 marks

4.2 Criteria for internal evaluation of Major Project (Phase I)

Internal evaluation is to be done by conducting a Viva by a team of evaluators comprising of the concerned guides and/or Head of the Department. The following are the components for internal evaluation of the **Major Project (Phase I):**

- ✓ Presentation of the work-5 marks
- ✓ Individual involvement & team work/ Attendance- 5marks
- ✓ Timely submission and assessment of 2 interim reports -10 marks

Total - 20 marks

OPEN COURSE I

AUCS581.a: CYBER SECURITY

Credit: 2 **Instructional Hours: 3 per week**

1. AIM

- To introduce the issues and methods of information security and its guidelines.

2. OBJECTIVES

On completion of this course student shall:

- Understand high-level overview of information security principles.
- Understand different roles and responsibilities of security professionals
- Understand cryptography and information system risk management.
- Be aware of multiple security control families as well as benefits of each control family

3. SYLLABUS

Module I: Introduction to Information systems: Modeling business process, components, categories, Individuals in information system, Developing information systems; **Information Systems:** threats, Information assurance, cyber security and security risk analysis; **Application security:** Data Security considerations, security technology, intrusion detection access control

Module II: Security threats: Introduction to security threats, Network and services attack, security threats to e-commerce

Module III: Security Policies: Introduction, Why we need security policies, Security policy development, Email security policies, Policy review process, corporate policy, sample template of cyber security policy

Module IV: Information security standards: Why ISO, IT ACT 2000, copyright, Patent, Intellectual property right, Cyber laws in India, Software Licensing, Semi conductor law and patent law,

4. REFERENCES

4.1 Core

- ***Fundamentals of Cyber security***, MayankBhushan, BPB publication, First Edition 2017

4.2 Assignments and activities:*Basic security for HTTP Applications and Services, Basic Security for SOAP Services*

NB:- Activities and assignments are not meant for End Semester Examination

OPEN COURSE I

AUCS581.b : DIGITAL MARKETING

Credit : 2 **Instructional Hours: 3 per week**

1. AIM

By the end of this course, participants will have had the opportunity to:

- understand the basics of digital marketing
- develop a comprehensive digital marketing strategy
- appreciate how to use new media such as mobile, search and social networking
- know the measurement techniques used in evaluating digital marketing efforts

2. OBJECTIVES

The emphasis in this course will be on developing fundamental concepts along with practical application. The objective of the course is:

- to help students to understand digital marketing methods
- to develop skills on engaging in marketing on the web using latest digital marketing techniques

3. SYLLABUS

Module I: Introduction to Digital Marketing: Overview of Digital Marketing Concepts, Personalized, Customized, Targeted and Contextual Ads , Differentiation with Traditional Marketing, Understanding various Social channels, Search Engine Optimization (SEO), Search Engine Marketing (SEM), PSA, PPM/CPM, PPC/ CPC, Google Ad words,Link Building, Google Algorithms , - Panda & Penguin, Directory Submissions.

Module II: Online Marketing: Email Marketing – Building Users Lists, Deliver Emails in Inbox & Generate Relevant Clicks. Social Media Marketing - Build Brand preferences, Generate Leads, Aggregate Audience, Mobile Marketing - App-based marketing, QR codes, Location-based marketing, SMS Marketing, Content and Blog Marketing – Info graphics, White paper, Brochure, Case studies, Word press

Module III: Web Analysis: Measure, Monitor and Evaluate the Website Traffic and Campaign Statistics, Introduction to Google Webmaster Tools, Integration of Google Webmaster Tools, Viewing and Understanding Reports, Drilling to Individual Keywords, Keywords Research and Analysis

Module IV: Website Monetization: Ways to generate revenues online - Blog writing, selling products, Affiliate programs, Conditional Advertising, Ad sales and Contextual Ads, Google AdSense.

4. REFERENCES

4.1 Core

- Paul Russell Smith, Dave Chaffey, ***E-Marketing Excellence: The Heart of E-Business***, Elsevier Butterworth-Heinemann, 2005, USA
- Judy Strauss, Raymond Frost, ***E-Marketing***, Prentice Hall, India

4.2 Internet Resources

- http://www.guru99.com/free-digital-marketing-tutorial.html
- http://www.omtac.com/h/n/omtac/digitalmarketingresi//292//

4.3 Assignments and activities: *Collection of current marketing tools, case studies, new trends.*

NB:- Activities and assignments are not meant for End Semester Examination

OPEN COURSE I

AUCS581.c : INTERNET AND WWW

Credit: 2 **Instructional Hours: 3 per week**

1. AIM

- To introduce to Internet and World Wide Web.

2. OBJECTIVES

- To understand the basic concepts of Networks.
- To learn the working of Internet.
- Exposure to Network Protocols and WWW.

3. SYLLABUS

Module I: Introduction to Network-Types of Network-Network Topologies, Intranet, Extranet-Advantages, Internet-History, Modes of Connecting to Internet-Dialup Access-Direct to dedicated connections, Internet Service Providers, Domain Name Service, Internet Addresses-Addressing Scheme-IPV6, Modems, Communication Software, Internet tool-File Transfer Protocols, Search Tools, Telnet.

Module II: Introduction to WWW and HTTP, Webpage, Introduction to Web Browser-Book Marks-Comparison, Directories, Search Engines-Search Fundamentals-Search Terminology-Search Strategies – Search Generalization-Search Specialization-Working.

Module III: Uniform Resource Locator (URL) Introduction to TCP/IP-TCP/IP Model, Email-Working with Email-Sending Mail-Reading Mail-Replying to Mail-Deleting Mail-Advantages and Disadvantages of Email, Basics of Chat Rooms, SMTP.

Module IV: Introduction to Web Server-Personal Web Server (PWS)-Internet Information Server (IIS)-Apache Webserver-Benefits of Web Server, Introduction to Security-Internet Security-Identifying Network Stations, Network Protocols-Internet Security Threats.

4. REFERENCES

4.1 Core

- Dr.SurenderJangra, "***Basics of Internet and Web***", Vayu Education of India. New Delhi 110002

4.2 Additional

- Raymond Greenlaw, Ellen Hepp, "***Fundamentals of Internet and the World Wide Web***, McGraw-Hill.

4.3 Assignments and activities:*Network Security , Security Requirements and Attacks, Confidentiality with Conventional Encryption, Message Authentication and Hash Functions, Public-Key Encryption and Digital Signatures, Secure Socket Layer and Transport Layer Security.*

NB:- Activities and assignments are not meant for End Semester Examination

ELECTIVE COURSE-I

AUCS591.a : DATA ANALYTICS

Credit: 3 **Instructional Hours: 4 per week**

1. AIM

- Understand the principles and purposes of data analytics, and articulate the different dimensions of the area.

2. OBJECTIVE

The student should be able to get the idea:

- To work with and manipulate a data set to extract statistics and features, coping with missing and dirty data.
- To get the basic knowledge of HADOOP
- To appreciate the need for privacy, identify privacy risks in releasing information, and design techniques to mediate these risks.

3. SYLLABUS

Module 1: Introduction, how analytics is used in practice, analytics works in different companies Google, Facebook, Kaggle, and Netflix.**BIG DATA** – Introduction, why big data – evolution principles- difference from regular data - convergence of key trends – A Wider Variety of Data - unstructured data – Big Data Business Models, Enabling Big Data Analytic Applications. **Big data**

analytics in industry – Digital Marketing and the Non - line World **-** web analytics – big data and marketing – New School of Marketing - fraud and big data – risk and big data – credit risk management – big data and algorithmic trading – big data and healthcare – big data in medicine – advertising and big data – big data technologies

Module II:Big Data Technology -Old vs. New Approaches, Data Discovery, Open-Source Technology, Cloud and Big Data, Predictive Analytics, Software as a Service, Mobile Business Intelligence, Crowdsourcing Analytics, Inter- and Trans-Firewall Analytics, R&D Approach Helps Adopt New Technology, Adding Big Data Technology into the Mix, **Information Management - B**ig Data Foundation, Big Data Computing Platforms, Big Data Computation, Big Data Storage, Big Data Computational Limitations, Big Data Emerging Technologies. **Business Analytics** - geospatial intelligence, Consumption of Analytics, Creation and Visualizing, Tools for Analytic Applications.

Module III:The People Part of the Equation - Evolution of Data Science, Learning over Knowing, Data Scientist Skills, Critical Thinking, Holistic View of Analytics, Setting Up the Right Organizational Structure forInstitutionalizing Analytics, **Data Privacy and Ethics -** Privacy Landscape, Customer Relationship Management, Rights and Responsibility, Technologies for anonymizing data,

Module IV:BASICS OF HADOOP - Introduction to Hadoop - Data, Data Storage and Analysis, Querying, Comparison with Other Systems- Relational Database Management Systems, Grid Computing, Volunteer Computing, A Brief History of Apache Hadoop. Design of Hadoop distributed file system (HDFS) – HDFS concepts – Java interface – data flow – Hadoop I/O – data integrity – compression – serialization, Avro – file-based data structures, HADOOP RELATED TOOLS - Hbase – data model and implementations – Hbase clients.

4. REFERENCES

4.1 Core

- Michael Minelli, Michelle Chambers, and AmbigaDhiraj, "***Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses***", Wiley, 2013.
- Tom White, "***Hadoop: The Definitive Guide***", Third Edition, O'Reilley, 2012

4.2 Assignments and activities:*DATA SHARING: Privacy, Anonymization, Risks - The ethics and risks of sharing data on individuals. Technologies for anonymizing data: k-anonymity, and differential privacy.*

ELECTIVE COURSE-I

AUCS591.b : MOBILE COMPUTING

Credit: 3 **Instructional Hours: 4 per week**

1. AIM

- To introduce Mobile Computing Techniques and Protocols

2. OBJECTIVES

- To understand the basic concepts of Mobile Computing.
- To learn the basics of mobile telecommunication
- Exposure to Ad-Hoc networks

3. SYLLABUS

Module I: Introduction- Mobile Computing- Mobile Computing Vs Wireless Networking-Mobile Computing Applications- Characteristics of Mobile Computing- Structure of Mobile Computing Applications. MAC Protocols-Wireless MAC issues-Fixed Assignment Schemes-Random Assignment Schemes-Reservation Based Schemes, 802.11 MAC Standard.

Module II: Global System for Mobile Communication (GSM)- GSM services, GSM Architecture, GSM Security, Mobile IP- Overview, Features Of Mobile IP- Key Mechanisms in Mobile IP, Route Optimization, Dynamic Host Configuration Protocol (DHCP), Overview of TCP/IP, Architecture Of TCP/IP, Improvement in TCP performance.

Module III: General Packet Radio Service (GPRS) – GPRS Services, GPRS Architecture, Universal Mobile Telecommunication System (UMTS)- UMTS Network Architecture, Operating Systems for Mobile Computing- Basic Concepts, Constraints and Requirements of Mobile OS, Commercial Mobile Operating Systems- Windows Phone, Palm OS, iOS, Android, Black Berry OS.

Module IV: Mobile Ad-Hoc Networks- Basic Concepts, Characteristics, Applications, Design Issues- Routing in MANET- Essentials of Traditional Routing Protocols, Popular MANET Routing Protocols, Vehicular Ad-Hoc Networks (VANET), MANET Vs VANET, Security, HDML and WAP (concepts only).

4. REFERENCES

4.1 Core

- Prasant Kumar Pattnaik, Rajib Mall, "***Fundamentals of Mobile Computing***", PHI Learning Pvt.Ltd., New Delhi 2012

4.2 Additional

- A K Talukder, H Ahmed, R RYavagal, ***Mobile Computing- Technology, Applications and Service Creation***, 2 ed., TMH.

4.3 Assignment and activities: *Case studies on different mobile OSs and programming languages, free and open source OS.*

NB:- Activities and assignments are not meant for End Semester Examination

ELECTIVE COURSE-I

AUCS591.c : TRENDS IN COMPUTING

Credit: 3 **Instructional Hours: 4 per week**

1. AIM

- Introduce advanced computing technologies and their application areas

2. OBJECTIVES

- To introduce the broad perceptive of cloud architecture& model
- To explore the fundamental concepts of big data analytics
- To introduce basics of edge computing and application
- How problems solved using soft computing

3. SYLLABUS

Module I: Cloud Computing: Technologies for network Based system-system models for distributed and cloud computing, Cloud Types, Cloud models- characteristics-cloud services (IaaS, PaaS, SaaS) – public vs. private cloud-computing on demand.

Module II: Data Storage in the cloud: Understanding , Advantages and Disadvantages of Cloud Based Data Storage; Service Oriented architecture- understanding SOA, web service; Implementing real time application over cloud platform.

Module III: Edge Computing: Overview, Edge computing terms and definitions, advantages, applications -grid computing: grid layered architecture, Distributed computing, mobile edge computing: introduction, reference architecture, application in 5G technologies.

Module IV: Soft computing: soft computing vs. hard computing: Introduction to Neural Network-Intelligence, Neurons, Artificial Neural network, Application scope of neural network, Brain vs computer. Problem areas, Training of artificial networks- supervised and Unsupervised; From ordinary set to fuzzy sets- Basics of Fuzzy set logic Theory, Foundation of fuzzy logic- fuzzy sets; Application.

4. REFERENCES

4.1Core

- ***Cloud Computing for Dummies*** by Judith Hurwitz, R.Bloor, M.Kanfman, F.Halper (Wiley India Edition)
- Venkata Krishna &etal, ***Principles of Grid computing - Concepts and application***, Ane Books
- Kris Jamsa, ***Cloud Computing***; Jones & Bartlett Lerning.
- Rahul Deva &GarimaKulshreshtha. ***Soft computing***. Shrof publishers & Distributers Pvt. Ltd.

4.2 Additional

- MichealBerkhold, David. j. Hand**,** "***Intelligent Data Analysis***", springer,2007
- ***Google Apps*** by Scott Granneman,Pearson
- ***Cloud Computing : A Practical Approach***, Antony T Velteet.al, McGraw Hill,
- ***Cloud Computing Bible*** by Barrie Sosinsky, Wiley India
- ***Edge computing IEEE journals and magazine***
- ***Data Analytics made accessible*** Dr. Anil Maheswari
- AnandRajaraman and Jeffry David Ullman "***mining of massive Datasets***" Cambridge university press
- RajkumarBuya and etal, ***Cloud computing – Principles and paradigm***, Wiley Publishers
- S. Rajasekaran&G.A.VijayalakshmiPai. ***Neural Networks. Fuzzy Logic and Genetic algorithms Synthesis and application***. PHI Learning Private Limited.
- Mahesh Mahajan &Rajdev Tiwari, ***Introduction to soft computing***, Acme Learning
- Dilip K Pratihar, ***soft computing – fundamentals &applications***,Narosa.
- S.N.Sivanandam, S.N. Deepa, ***Principles of soft computing***,Wiley India.

4.3 Assignment and activities: *Study of cloud computing, Migrating to cloud, revolutionary development in edge computing, GPS application, Hadoop technology, Engineering, Industries, government application of Big data*

NB:- Activities and assignments are not meant for End Semester Examination

SEMESTER VI

CORE COURSE XXVIII

AUCS641: DATA MINING AND WAREHOUSING

Credit: 4 **Instructional Hours: 4 per week**

1. AIM

- To get an entry level understanding of the concepts of data mining

2. OBJECTIVES

- To get an understanding of the general properties of data in large databases
- Understand a variety of real-world applications that require data mining
- How to discover useful patterns and associations in huge quantities of data

3. SYLLABUS

Module I: Introduction:-Data, Information, Knowledge, KDD, types of data for mining, Application domains, data mining functionalities/tasks. Data processing—Understanding data, pre-processing data-Form of data processing, Data cleaning (definition and Phases only), Need for data integration, Steps in data transformation, Need of data reduction

Module II: Data Warehouses-Databases, Data warehouses, Data Mart, Databases Vs Data warehouses, Data ware houses Vs Data mart, OLTP OLAP, OLAP operations/functions, OLAP Multi-Dimensional Models- Data cubes, Star, Snow Flakes, Fact constellation. Association rules-Market Basket Analysis, Criteria for classifying frequent pattern mining, Mining Single Dimensional Boolean Association rule-A priori algorithm

Module III: Classification- Classification Vs Prediction, Issues, Decision trees, Bayes classification-Bayes Theorem, Naïve Bayesian classifier, K Nearest Neighbour method, Rule-Based classification - Using IF…THEN rules for classification

Module IV: Cluster analysis: definition and Requirements, Characteristics of clustering techniques, Types of data in cluster analysis, categories of clustering-Partitioning methods, K-Mean and K - method only, outlier detection in clustering.

4. REFERENCES

4.1 Core

- Sunitha Tiwari &NehaChaudary, ***Data Mining and Warehousing***, DhanpatRai& Co.

4.2 Additional

- Jiawei Han &MichelineKamber&Jian Pei ***Data Mining Concepts & Techniques***
- Margaret H Dunham ***Data Mining-Introductory& Advanced Topics***; Pearson

4.3 Assignment and activities: *Mining web, temporal, text, multimedia, medical data and other Applied Data mining areas; OLAP tools; Introduction to Rapid Miner and other free and open source data mining tools.*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE XXIX

AUCS642 : INFORMATION SECURITY

Credit: 4 **Instructional Hours: 4 per week**

1. AIM

- To introduce internetworking and the issues and methods of information security over internetworks.

2. OBJECTIVES

On completion of this course student shall

- be aware of principles and protocols of internetworks
- understand the basic issues in information security
- understand the concept of ciphers and cryptography
- to impart an idea on various ciphers
- understand the concept of digital signatures and e-mail security policies
- impart an idea on malicious software and remedies

3. SYLLABUS

Module I: Information Security: Network security, Confidentiality, integrity, authentication, security policy, basic network security terminology, cryptography, symmetric encryption, substitution ciphers, transposition ciphers, steganography, Block ciphers, modes of operation, Data Encryption Standard, Public key cryptography, applications, strength and weakness, RSA algorithm, key distribution (concepts only).

Module II: Authentication: authentication methods, message digest, digital signatures, digital signature algorithm, DSS, E-mail security: Pretty Good Privacy, working of PGP, S/MIME, MIME, IP Security, Architecture, IPSec: strengths and benefits, IPv4, IPv6, ESP protocol, Web Security: Secure Socket layer, SSL session and connection.

Module III: Malicious Software: viruses, working of anti-virus software, worms, Trojans, spyware, firewall, characteristics of firewall, packet filters, application level gateways, firewall architecture, trusted systems.

Module IV: Security and Law: Regulations in India. Information Technology Act 2000/2008.Cyber Crime and the IT Act 2000/2008.Indian Contract Act 1872, Indian Penal Code, Indian Copyright Act, Consumer Protection Act.Future Trends – The Law of Convergence

4. REFERENCES

4.1 Core

- Pachghare V. K., ***Cryptography and Information Security,*** PHI.
- Brijendra Singh, ***Cryptography & Network Security***, PHI.

4.2 Additional References

- William Stallings, ***Cryptography and Network Security: Principles and Practice***, Pearson Education India

4.3 Assignment and activities: *Wireless security, User access technologies, Wired: xDSL, FTTH, Cellular wireless: GPRS, EDGE, HSPDA, Broadband wireless: 802.16.*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE XXX

AUCS643 : ARTIFICIAL INTELLIGENCE

Credit: 4 **Instructional Hours: 4 per week**

1. AIM

- To Expose students to basic concepts and tools of Artificial intelligence and create awareness about its applications, both current and futuristic.

2. OBJECTIVES

- To introduce the notion of machine intelligence.
- To introduce the Symbolic processing paradigm of AI.
- To introduce Knowledge representation formalism.
- To introduce basic concepts and challenges of Speech and Language Processing.
- To introduce basic concepts and challenges of Expert Systems.

3. SYLLABUS

Module I: Overview of Artificial Intelligence: What is AI, The importance of AI; Knowledge: Introduction, Definition and Importance of knowledge, Knowledge–Based Systems, Representation of Knowledge, Knowledge Organization, Knowledge Manipulation, Acquisition of Knowledge.

Module II: Formalized Symbolic Logics: Introduction, Syntax and Semantics for Propositional Logic and FOPL, Properties of Wffs, Conversion to Clausal Form, Inference Rules, The Resolution Principle; Structured Knowledge: Associative Networks, Frame Structures, Conceptual Dependencies and Scripts.

Module III: Search and Control Strategies: Preliminary concepts, Examples of Search Problems, Uniformed or blind Search, Informed Search, Searching And-Or graphs; Matching Techniques: Introduction, Structures Used in Matching, Measures for Matching, Partial Matching, The RETE Matching Algorithm.

Module IV: Natural Language Processing : Introduction, Overview of Linguistics, Grammars and Languages, Basic Parsing Techniques, Semantic Analysis and Representation Structures, Natural Language Generation, Natural Language Systems; Expert Systems : Introduction, Rule Based System Architecture, Knowledge Acquisition and Validation, Knowledge System Building Tools.

4. REFERENCES

4.1 Core

- Introduction to ***artificial intelligence and expert systems***, DAN W. PATTERSON, PHI Learning 2014

4.2 Additional

- ***Artificial Intelligence***, Third Edition, Elaine Rich, Kevin Knight, Shivashankar B Nair, McGraw Hill Education (India) PVT LTD

4.3 Assignment and activities:*Generic Introduction to LISP and PROLOG, Critic of Artificial Intelligence, Neural Network Models, Future of AI.*

NB:- Activities and assignments are not meant for End Semester Examination

CORE COURSE XXXI

AUCS644 : MAJOR PROJECT PHASE II

Credit: 4 **Instructional Hours: 9 per week**

1. AIM

- To expose students to industry-standard project practices, through a project work under time and deliverable constraints, applying the knowledge acquired through various courses.

2. OBJECTIVES

- To provide an opportunity to apply the knowledge gained through various courses .
- To provide an opportunity to practice different phases of software/system development life cycle

- To introduce the student to a professional environment and/or style typical of a global IT industry
- To provide an opportunity for structured team work and project management
- To provide an opportunity for effective, real-life, technical documentation
- To provide an opportunity to practice time, resource and person management.

3. PROJECT GUIDELINES

- Group Size – Maximum 4, most preferably- 3
- No. of records – No. of group members + 1 (Department copy)
- Certificate should include the names of all members

The minimal phases for the project are:

- Project search
- finalization and allocation
- Investigation of system requirements
- Data and Process Modelling
- System Design
- Program design
- Program coding and unit testing
- System integration
- System implementation and acceptance testing

3.1 Planning the Project:

The Major Project is an involved exercise which has to be planned well in advance. The topic should be chosen in Semester 4. Related reading, training and discussions should start from semester 5 itself.

3.2 Selection of Project Work:

Project work could be of 3 types:

a) Developing solution for a real-life problem: In this case, a computer involved solution for an existing real life problem is recommended. Examples are Accounting Software Package for a particular organization, Computerisation of administrative functions of an organization, Web Based Commerce, etc. The scope for creativity and exploration in such projects are limited, but if done meticulously, valuable experience in industrial context can be gained.

(b) Innovative Product development: These are projects where a clear-cut computer based solution may not be existing but a possible utility for the same is conceived by the proposer. An Example is a Malayalam Language Editor with Spell Checker, Computer Music Software for Indian Music, Heat Engines Simulation Software for eLearning, Digital Water Marking Software etc.

(c) Research level project: These are projects which involve research and development and may not be as structured and clear cut as in the above case. Examples are Malayalam Character Recognition, Neural Net Based Speech Recogniser, Biometric Systems, and Machine Translation System etc. These projects provide more challenging opportunities to students, but at EX level is a difficult

choice. If any student identifies proper support in terms of guidance, technology and references from external organizations and also the supervisors are convinced of the ability of the student(s) to take up the project, it shall be permitted. The methodology and reporting of such projects could be markedly different from type (a) and is left to the proposer/external supervisor of the projects.

3.3 Selection of Team:

To meet the stated objectives, it is imperative that Major Project is done through a team effort**.** Though it would be ideal to select the team members at random (drawing lots) and this should be strongly recommended, due to practical considerations, students may also be given the choice of forming themselves into teams of 3 members(teams less than 3 members may be permitted in Exceptional cases, for valid reasons). A gender mix should also be strongly suggested. A team leader shall be elected through drawing lots. Teams shall maintain team meeting minutes and ensure that every team member has tasks assigned in writing. Team meeting minutes shall form a part of the Project Report. Even if students are doing projects as groups, each one must independently take up different modules of the work and must submit the reports also independently (though, in such cases, some common materials are permissible). Evaluation will also be done independently.

3.4 Selection of Tools:

No restrictions shall be placed on the students in the choice of platforms/tools/languages to be utilized for their project work, though open source is strongly recommended, wherever possible. No value shall be placed on the use of tools in the evaluation of the project.

3.5 Selection of Organisation & Guide:

No restrictions shall be placed on the students in the choice of organization where project work may be done, in terms of locality, type (public/private) etc. It is the duty of the Head of Institute/Principal of College to ensure that the Aim, Objectives and full project guidelines are communicated to the external organization. The guide should ideally be a post-graduate with minimum 2 years of work experience.

Students may also choose to do project in the college/institute (or partially in the college/institute and partially in an external organization), especially product-based work, but in such cases the supervisors must ensure that (i) industry practices are followed (ii) the students undertake a planned visit to an IT industry with international operations to make up for the loss of experience and (iii) the services of an external guide with industry experience is obtained.

3.6 Project Management:

Head of the Department should publish a list of students, projects topics, internal guide and external organization (if any) and teams agreed, at the beginning of semester 5. Changes in this list may be permitted for valid reasons and shall be considered favourably by Head of the Department any time before commencement of the project. Any request for change after commencement should considered by a committee of 3 teachers and their recommendation shall be accepted by Head of the Department.

Gantt-chart of proposed activities and a draft statement of project deliverables (which may subsequently be altered if justified) should be prepared before the commencement of the project. The actual completion of each phase should be noted on the chart in the course of the project work. Students should submit a fortnightly report of progress which could be indication of percentage of completion marked on the original Gantt-chart, with any notes attached. Students should ideally keep a daily activity log sheet. Team meetings should be documented in the format given at the end. Changes in the submitted documents are possible, as project development is essentially an evolutionary process. The project guide must ensure that changes are necessary due to the knowledge gained in succeeding phases of the project. The date of completion of a phase should be brought forward if the changes made are deemed to be errors and not due to additional knowledge gained from a succeeding phase.

3.7 Documentation:

The following are the major guidelines:

- The final outer dimensions of the report shall be 21 cm X 30 cm.
- The colour of the flap cover shall be light green.
- Only hard binding should be done, with title of the Project and the words "< TITLE>BSc(Computer Science) Project Report 2__" displayed on the spine in 20 point, Bold, Times New Roman.
- It is highly recommended that Latex be used for documentation.
- The text of the report should be set in 12 pt, Times New Roman, 1.5 Spaced.
- Headings should be set as follows: CHAPTER HEADINGS 20 pt, Times New Roman, Bold, All Caps, Centered.
- 1. SECTION HEADINGS 12 pt, Times New Roman, Bold, All Caps, Left Adjusted.
- 1. 1 Section Sub-headings 12 pt, Times New Roman, Bold, Left Adjusted.
- Titles of Figures, Tables etc are done in 12 point, times New Roman, Italics, Centered.

<PROJECT TITLE>

<STUDENT'S NAME>

<COLLEGE NAME>

<DEPARTMENT NAME>

<COLLEGE LOGO>

PROJECT REPORT

Submitted in partial fulfilment of the

Requirements for the award of

B. Sc. Computer Science degree

20_ _

Some general guidelines on documentation stylistics are:

- Double quotes and single quotes (“”, ‘’) should be used only when essential. In most cases words put in quotes are better highlighted by setting them in italics. Eg: This process is known as “morphing”. This process is known as *morphing*.
- Page numbers shall be set at right hand top corner
- Paragraph indent shall be set as 3.
- Only single space need be left above a section or sub-section heading and no space may be left after them.
- Certificate should be in the format:

<COLLEGE NAME>

<DEPARTMENT NAME>

<COLLEGE LOGO>

“Certified that this report titled.....is a bonafiderecord of the project work done by Sri/Kum.....under our supervision and guidance, towards partial fulfilmentof the requirements for the award of the Degree of B. Sc. Computer Science”.

Signature of Internal Guide **Signature of External Examiner**

Head of the Department

- If the project is done in an external organization, another certificate on the letterhead of the organization is required:

External Organization Letter Head

“Certified that this report titled..... is a bonafiderecord of the project work done by Sri/Kum.....under our supervision and guidance, at the .....Department of..... (Organization) towards partialfulfilment of the requirements for the award of the Degree of B.Sc. Computer Science”.

Signature

- References shall be in IEEE format (see any IEEE magazine or transaction). Take care in use of italics and punctuation.While doing the project, keep note of all the books you refer, in the correct format, and include them in alphabetical order in your reference list. Eg: A book is cited as: Kartalopoulos S. V., Understanding Neural Networks and Fuzzy Logic, BPB Publishers, 1996, pp. 21-27. (pp.21-27 indicates that pages 21-27 have been referred. If the whole book is being referred, this may be omitted. If a single page is referred, say 7, it may be cited as p.7.

- Report writing is NOT a hasty activity done after finishing the project. Students must try to develop the report along with the work, so as to give it flesh and blood.
- Drafts should be read, modified, spell checked and grammar checked at least thrice during the course of the project and before a final printout is taken, the same may be got approved from the internal guide.
- The students should send two interim reports after the analysis and design phases of the project to internal guides. This will also help the students in their report writing.
- A soft copy of the complete documentation, including source code, should be maintained for any clarification during assessments.
- The Gantt chart, fortnightly progress reports, and team meeting minutes mentioned in section 3.6 should appear as appendix to the project report.
- Regarding the body of the report, as an indicative Example, the following is given (though students should not attempt to fit every kind of project report into this format):
  - ✓ Organizational overview (of the client organization, where applicable)
  - ✓ Description of the present system
  - ✓ Limitations of the present system
  - ✓ The Proposed system- Its advantages and features
  - ✓ Use Case diagram of the proposed system.
  - ✓ Top level DFD of the proposed system with at least one additional level of Expansion
  - ✓ Structure Chart of the System
  - ✓ System flowchart
  - ✓ Files or tables (for DBMS projects) list. Class names to be entered for each file in OO systems.
  - ✓ List of fields or attributes (for DBMS projects) in each file or table.
  - ✓ Program – File table that shows the files/tables used by each program and the files are read, written to, updated, queried or reports were produced from them.
  - ✓ Reports List with column headings and summary information for each report.
  - ✓ System Coding and variable/file/table naming conventions
  - ✓ System controls and standards
  - ✓ Screen layouts for each data entry screen.
  - ✓ Report formats for each report.
- Program documentation is suggested on the following lines:
  - ✓ Program id
  - ✓ Program function Explanation
  - ✓ Data entry screen (reproduced from system documentation).
  - ✓ Report layout (reproduced from system documentations)
  - ✓ Program level pseudo code or flowchart.
  - ✓ Decision tables, decision trees with explanation in English wherever necessary.
  - ✓ Program listing
  - ✓ Test data
  - ✓ Test results

3.8 Methodology

Wherever applicable, object oriented approach should be used for software development. The project report should generally contain details of the following steps *(though students should not attempt to fit every kind of project into this format):*

(a) Analysis

- ✓ Study of existing systems and its drawbacks (general)
- ✓ Understanding the functionalities of the system (detailed)
- ✓ Preparation of requirement
- ✓ Conduct of Feasibility study
- ✓ Identification of relevant Objects
- ✓ Abstraction of each object (attributed and methods)
- ✓ Relationship between objects

(b) Design

- ✓ Design of each subsystems
- ✓ Design of each classes
- ✓ Design of communications between objects
- ✓ Design of Algorithms for problem solving
- ✓ User interface Design
- ✓ Any other steps if necessary

(c) Coding and Implementation

(d) Testing

(e) Security, Backup and Recovery Mechanisms

(f) On line help and User Manuals

(g) Upgradability Possibilities

3.9 Project IPR & Utilisation:

The intellectual property rights in all project work done by the students shall vest with the college, except in cases where some external organizations seek undertaking from students to concede IPR in all work done in their organization or under their guidance. Where possible, students should attempt to obtain at least a joint IPR for the College. In cases where project works are of public utility, students shall be asked to publish their work including source code and documentation, in so far as their rights are clear.

4. REFERENCES

4.1 Core

- S. A. Kelkar, Software Project Management, Prentice Hall of India
- W. Alan Randolph, Barry Z. Posner, ***Effective project planning and management***, PHI

4.2 Additional

- Greg Mandanis, ***Software Project Management Kit for Dummies*****,** IDG Books
- Joel Henry, ***Software Project Management: A Real - World Guide to Success***, Pearson Education

- Frederic P. B., ***The Mythical Man-Month: Essays on Software Engineering***, Addison Wesley
- David Lamport, ***Latex: A document Preparation System***, 2/e, Pearson Education

5. EVALUATION

5.1 Criteria for external evaluation of Major Project

External evaluation is done by an external examiner appointed by the College. The following components are to be assessed for the End Semester Evaluation of the Major Project:

- ✓ Documentation- 30 marks
- ✓ Presentation of work- 25 marks
- ✓ Viva - 25 marks

Total - 80 marks

5.2 Criteria for internal evaluation of Major Project

Internal evaluation is be done by conducting a viva voce by a team of evaluators comprising of the concerned guides and/or Head of the Department. The following are the components for internal evaluation of the Major Project:

- ✓ Presentation of the work-5 marks
- ✓ Individual involvement & team work/ Attendance- 5marks
- ✓ Timely submission and assessment of 2 interim reports -10 marks

Total - 20 marks

ELECTIVE COURSE-II

AUCS691.a : COMPUTATIONAL LINGUISTICS

Credit: 4 **Instructional Hours: 4 per week**

1. AIM

- To learn fundamental concepts and algorithms in computational linguistics and natural language processing
- To learn Malayalam Computing

2. OBJECTIVES

On completion of this course student shall

- understand the basic concepts in computational linguistics
- understand the basic concepts in natural language processing
- familiar with Malayalam computing & typing tools

3. SYLLABUS

Module I: What is linguistics?, words &morphemes, inflection and derivation; exceptions, consonants andvowels,

phonetics,Morphology,Syntax,Semantics,Pragmatics,Phonology,Sociolinguistics,Historical and Comparative Linguistics,Psycholinguistics,Sociology of Language. Relevance of linguistics to other fields of enquiry—Philosophy, Anthropology, Sociology, Political Science, Psychology, Education, Computer Science and Literature.

Module II: Socio-linguistics and Applied Linguistics,Language and Society: Speech community; verbal repertoire; linguistic and communicative competence; linguistic variability and ethnography of speaking; socio- linguistic variables; patterns of variation; regional, social and stylistic; restricted and elaborated codes; diglossia. Corpus Linguistics,Neurolinguistics,UNICODE,difference between Unicode and ASCII, History of Unicode System,UNICODE and FOSS

Module III: Language Technology: Natural language interface to databases, Speech technology—text-to-speech and speech-to-text systems, Machine (aided) translation; computer aided language teaching; text processing;Malayalam computing Tools-Unicode Fonts-Installation of Fonts ,Malayalam typing tools- Phonetic Vs In script Swanalekha -Installation and uses ,Need of Font Converter(ASCII to Unicode).

Module IV: Natural Language Processing-Morphology: Scope and nature: concept of morpheme, morph, lexeme and word; identification of morphemes; morphological alternation; morphophonemic process; internal and external sandhi; derivation vs inflection; root and stem; grammatical categories—tense, aspect, mood, person, gender, number, case; case marker and case relation; pre- and post-positions; stem vs word-based morphology;Malayalam morphological analysis and generation,Part of speech tagging,Parsing,DiscourseAnalysis,Anaphora Resolution.

4. REFERENCES

- Allen, James, ***Natural Language Understanding***, Second Edition, Benjamin/Cumming, 1995.
- Charniack, Eugene, ***Statistical Language Learning***, MIT Press, 1993.
- Jurafsky, Dan and Martin, James, ***Speech and Language Processing***, Second Edition, Prentice Hall, 2008.
- Manning, Christopher and Heinrich, Schutze, ***Foundations of Statistical Natural Language Processing***, MIT Press, 1999.

4.1 Assignment and activities: *Case study on Swathanthra Malayalam Computing*

NB:- Activities and assignments are not meant for End Semester Examination

ELECTIVE COURSE-II

AUCS691.b : CYBER FORENSICS

Credit: 4 **Instructional Hours: 4 per week**

1. AIM

- To learn computer forensics
- To become familiar with forensics tools
- To learn to analyze and validate forensics data

2. OBJECTIVES

- Understand the basics of computer forensics
- Apply a number of different computer forensic tools to a given scenario
- Analyze and validate forensics data
- Identify the vulnerabilities in a given network infrastructure
- Implement real-world hacking techniques to test system security

3. SYLLABUS

Module I: Introduction to computer forensics: Introduction to Traditional Computer Crime, Traditional problems associated with Computer Crime. Introduction to Identity Theft & Identity Fraud. Types of CF techniques – Incident and incident response methodology – Forensic duplication and investigation. Preparation for IR: Creating response tool kit and IR team. – Forensics Technology and Systems – Understanding Computer Investigation – Data Acquisition.

Module II:Evidence collection and forensics tools: Processing Crime and Incident Scenes – Working with Windows and DOS Systems. Current Computer Forensics Tools: Software/ Hardware Tools.

Module III: Analysis and validation: Validating Forensics Data – Data Hiding Techniques – Performing Remote Acquisition – Network Forensics – Email Investigations – Cell Phone and Mobile Devices Forensics.

Module IV: Ethical hacking: Introduction to Ethical Hacking – Footprinting and Reconnaissance – Scanning Networks – Enumeration – System Hacking – Malware Threats – Sniffing. **Ethicalhacking in web:** Social Engineering – Denial of Service – Session Hijacking – Hacking Web servers – Hacking Web Applications – SQL Injection – Hacking Wireless Networks – Hacking Mobile Platforms.

4. REFERENCES

4.1 Core

- Bill Nelson, Amelia Phillips, Frank Enfinger, Christopher Steuart, ***—Computer Forensics and Investigations***, Cengage Learning, India Edition, 2016.
- ***CEH official Certified Ethical Hacking Review Guide***, Wiley India Edition, 2015.

4.2 Additional

- John R.Vacca, ***—Computer Forensics***, Cengage Learning, 2005
- MarjieT.Britz, ***—Computer Forensics and Cyber Crime: An Introduction***, 3rd Edition, Prentice Hall, 2013.
- AnkitFadia ***— Ethical Hacking*** Second Edition, Macmillan India Ltd, 2006
- Kenneth C.Brancik ***—Insider Computer Fraudn***Auerbach Publications Taylor & Francis Group–2008.

4.3 Assignment and activities: *Highlight the key steps to be performed in solving a computer forensic case; The key organizational guidelines on cell phone forensics; Ethical Dimension of Cybercrimes; Intellectual Property in the Cyberspace*

NB:- Activities and assignments are not meant for End Semester Examination

ELECTIVE COURSE-II

AUCS691.c : FREE AND OPEN SOURCE SOFTWARE

Credit: 4 **Instructional Hours: 4 per week**

1. AIM

- To introduce different free OS

2. OBJECTIVES

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

- Explain the features of free & open source software
- Familiarization with LINUX
- Familiarization with Shell Scripting

3. SYLLABUS

Module I: Introduction to FOSS –Software – System Software – Application software – Proprietary software –Free Software – Open Source Software - The FOSS Philosophy – FOSS development method – History of FOSS – Benefits of FOSS

Module II: Linux – Linux as the Kernel – Linux as a distribution – License agreement for FOSS – Localization and Internalization – Free Software and GNU - Methods for localizing GNU/Linux

Module III: Ubuntu Basics – Introduction – Ubuntu Releases – Ubuntu versions - The Gnome Display Manager: GDM – The Ubuntu Desktop – Ubuntu Administrative Tools -Software Management with DEB, APT, and dkpg –Display Drivers –The File Structure

Module IV: Shell Scripting:The Shell – Definition - Accessing Shells – The Command Line – History – Filename Expansion – Standard Input/Output and Redirection – Pipes – Redirecting and Piping the standard Error - Jobs: Background, Kills, and Interruptions - Ending Processes: ps and kill - Shell Variables - Shell Scripts\: User-Defined Commands - Control Structures - Filters and Regular

Expressions – Aliases - Configuring Your Shell with Shell Parameters – Listing, Displaying, and Printing Files: ls, cat, more, less, and lpr – Managing Directories: mkdir, rmdir, ls, cd, and pwd - Controlled Administrative Access: PolicyKit, sudo, and gksu - Managing Software with APT - Managing Non-repository Packages with dpkg.

4. REFERENCES

4.1 Core

- Richard Petersen - ***Ubuntu The Complete Reference***, McGraw Hill

4.2 Additional

- Kenneth Wong and PhetSayo - ***Free/Open Source Software : A general Introduction***, UNDP-APDIP, 2004

4.3 Assignment and activities: *Case studies on different Free Operating Systems.*

NB:- Activities and assignments are not meant for End Semester Examination

****