**VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI**
**Scheme of Teaching and Examination 2017-2018**
**Choice Based Credit System (CBCS)**

## B.E: Computer Science and Engineering

**III SEMESTER**

| Sl. No | Course Code | Title | Teaching Department | Teaching Hours /Week | | Examination | | | | Credits |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | Theory | Practical/ Drawing | Duration in hours | SEE Marks | CIE Marks | Total Marks | |
| 1 | 17MAT31 | Engineering Mathematics - III | Maths | 04 | | 03 | 60 | 40 | 100 | 4 |
| 2 | 17CS32 | Analog and Digital Electronics | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 3 | 17CS33 | Data Structures and Applications | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 4 | 17CS34 | Computer Organization | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 5 | 17CS35 | Unix and Shell Programming | CS/IS | 03 | | 03 | 60 | 40 | 100 | 3 |
| 6 | 17CS36 | Discrete Mathematical Structures | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 7 | 17CSL37 | Analog and Digital Electronics Laboratory | CS/IS | 01-Hour Instruction 02-Hour Practical | | 03 | 60 | 40 | 100 | 2 |
| 8 | 17CSL38 | Data Structures Laboratory | CS/IS | 01-Hour Instruction 02-Hour Practical | | 03 | 60 | 40 | 100 | 2 |
| 9 | 17KL/CPH39/49 | Kannada/Constitution of India, Professional Ethics and Human Rights | Humanities | 01 | | 01 | 30 | 20 | 50 | 01 |
| **TOTAL** | | | | **Theory: 24hours Practical: 06 hours** | | **25** | **510** | **340** | **850** | **28** |

**1.Kannada/Constitution of India, Professional Ethics and Human Rights:** 50 % of the programs of the Institution have to teach Kannada/Constitution of India, Professional Ethics and Human Rights in cycle based concept during III and IV semesters.

**2. Audit Course:**
(i) *All lateral entry students (except B.Sc candidates) have to register for Additional Mathematics – I, which is 03 contact hours per week.

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 17MATDIP31 | Additional Mathematics –I | Maths | 03 | | 03 | 60 | – | 60 | – |

(ii) Language English (Audit Course) be compulsorily studied by all lateral entry students (except B.Sc candidates)



H.O.D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI**
**Scheme of Teaching and Examination 2017-2018**
**Choice Based Credit System (CBCS)**

## B.E: Computer Science and Engineering

**IV SEMESTER**

| Sl. No | Course Code | Title | Teaching Department | Teaching Hours /Week | | Examination | | | | Credits |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | Theory | Practical/ Drawing | Duration in hours | SEE Marks | CIE Marks | Total Marks | |
| 1 | 17MAT41 | Engineering Mathematics - IV | Maths | 04 | | 03 | 60 | 40 | 100 | 4 |
| 2 | 17CS42 | Object Oriented Concepts | CS/IS | 03 | | 03 | 60 | 40 | 100 | 3 |
| 3 | 17CS43 | Design and Analysis of Algorithms | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 4 | 17CS44 | Microprocessors and Microcontrollers | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 5 | 17CS45 | Software Engineering | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 6 | 17CS46 | Data Communication | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 7 | 17CSL47 | Design and Analysis of Algorithm Laboratory | CS/IS | 01-Hour Instruction 02-Hour Practical | | 03 | 60 | 40 | 100 | 2 |
| 8 | 17CSL48 | Microprocessors Laboratory | CS/IS | 01-Hour Instruction 02-Hour Practical | | 03 | 60 | 40 | 100 | 2 |
| 9 | 17KL/CPH39/49 | Kannada/Constitution of India, Professional Ethics and Human Rights | Humanities | 01 | | 01 | 30 | 20 | 50 | 01 |
| | | | **TOTAL** | **Theory: 24hours Practical: 06 hours** | | **25** | **510** | **340** | **850** | **28** |

**1. Kannada/Constitution of India, Professional Ethics and Human Rights:** 50 % of the programs of the Institution have to teach Kannada/Constitution of India, Professional Ethics and Human Rights in cycle based concept during III and IV semesters.

**2.Audit Course:**
(i) *All lateral entry students (except B.Sc candidates) have to register for Additional Mathematics – II, which is 03 contact hours per week.

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 17MATDIP41 | Additional Mathematics –II | Maths | 03 | | 03 | 60 | -- | 60 | -- |

(ii) Language English (Audit Course) be compulsorily studied by all lateral entry students (except B.Sc candidates)

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI**
**Scheme of Teaching and Examination 2017-2018**
**Choice Based Credit System (CBCS)**

**B.E: Computer Science and Engineering**

**V SEMESTER**

| Sl. No | Course Code | Title | Teaching Department | Teaching Hours /Week | | Examination | | | | Credits |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | Theory | Practical/ Drawing | Duration in hours | SEE Marks | CIE Marks | Total Marks | |
| 1 | 17CS51 | Management and Entrepreneurship for IT Industry | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 2 | 17CS52 | Computer Networks | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 3 | 17CS53 | Database Management System | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 4 | 17CS54 | Automata theory and Computability | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 5 | 17CS55x | Professional Elective-1 | CS/IS | 03 | | 03 | 60 | 40 | 100 | 3 |
| 6 | 17CS56x | Open Elective-1 | CS/IS | 03 | | 03 | 60 | 40 | 100 | 3 |
| 7 | 17CSL57 | Computer Network Laboratory | CS/IS | 01-Hour Instruction 02-Hour Practical | | 03 | 60 | 40 | 100 | 2 |
| 8 | 17CSL58 | DBMS Laboratory with mini project | CS/IS | 01-Hour Instruction 02-Hour Practical | | 03 | 60 | 40 | 100 | 2 |
| | | | **TOTAL** | **Theory: 22hours Practical: 06 hours** | | **24** | **480** | **320** | **800** | **26** |

| **Professional Elective-1** | | | **Open Elective – 1*** (List offered by CSE Board only)** | |
|---|---|---|---|---|
| 17CS551 | Object Oriented Modeling and Design | | 17CS561 | Programming in JAVA *(Not for CSE/ISE students)* |
| 17CS552 | Introduction to Software Testing | | 17CS562 | Artificial Intelligence |
| 17CS553 | Advanced JAVA and J2EE | | 17CS563 | Embedded Systems |
| 17CS554 | Advanced Algorithms | | 17CS564 | Dot Net framework for application development; |
| | | | 17CS565 | Cloud Computing *(Not for CSE/ISE students)* |

***Students can select any one of the open electives offered by any Department (Please refer to consolidated list of VTU for open electives).
Selection of an open elective is not allowed, if:
· The candidate has no pre – requisite knowledge.
· The candidate has studied similar content course during previous semesters.
· The syllabus content of the selected open elective is similar to that of Departmental core course(s) or to be studied Professional elective(s).
Registration to open electives shall be documented under the guidance of Programme Coordinator and Adviser.

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI**
**Scheme of Teaching and Examination 2017-2018**
**Choice Based Credit System (CBCS)**

**B.E: Computer Science and Engineering**

**VI SEMESTER**

| Sl. No | Course Code | Title | Teaching Department | Teaching Hours /Week | | Examination | | | | Credits |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | Theory | Practical/ Drawing | Duration in hours | SEE Marks | CIE Marks | Total Marks | |
| 1 | 17CS61 | Cryptography, Network Security and Cyber Law | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 2 | 17CS62 | Computer Graphics and Visualization | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 3 | 17CS63 | System Software and Compiler Design | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 4 | 17CS64 | Operating Systems | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 5 | 17CS65x | Professional Elective-2 | CS/IS | 03 | | 03 | 60 | 40 | 100 | 3 |
| 6 | 17CS66x | Open Elective-2 | CS/IS | 03 | | 03 | 60 | 40 | 100 | 3 |
| 7 | 17CSL67 | System Software and Operating System Laboratory | CS/IS | 01-Hour Instruction 02-Hour Practical | | 03 | 60 | 40 | 100 | 2 |
| 8 | 17CSL68 | Computer Graphics Laboratory with mini project | CS/IS | 01-Hour Instruction 02-Hour Practical | | 03 | 60 | 40 | 100 | 2 |
| | | | **TOTAL** | **Theory:22hours Practical: 06 hours** | | **24** | **480** | **320** | **800** | **26** |

| **Professional Elective-2** | | | **Open Elective – 2*** (List offered by CSE Board only)** | |
|---|---|---|---|---|
| 17CS651 | Data Mining and Data Warehousing | | 17CS661 | Mobile Application Development |
| 17CS652 | Software Architecture and Design Patterns | | 17CS662 | Big Data Analytics *(Not for CSE/ISE students)* |
| 17CS653 | Operations research | | 17CS663 | Wireless Networks and Mobile computing |
| 17CS654 | Distributed Computing system | | 17CS664 | Python Application Programming |
| | | | 17CS665 | Service Oriented Architecture |
| | | | 17CS666 | Multicore Architecture and Programming |

***Students can select any one of the open electives offered by any Department (Please refer to consolidated list of VTU for open electives).
Selection of an open elective is not allowed, if:
· The candidate has no pre – requisite knowledge.
· The candidate has studied similar content course during previous semesters.
· The syllabus content of the selected open elective is similar to that of Departmental core course(s) or to be studied Professional elective(s).
Registration to open electives shall be documented under the guidance of Programme Coordinator and Adviser.



H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI**
**Scheme of Teaching and Examination 2017-2018**
**Choice Based Credit System (CBCS)**

## B.E: Computer Science and Engineering

**VII SEMESTER**

| Sl. No | Course Code | Title | Teaching Department | Teaching Hours /Week | | Examination | | | | Credits |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | Theory | Practical/ Drawing | Duration in hours | SEE Marks | CIE Marks | Total Marks | |
| 1 | 17CS71 | Web Technology and its applications | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 2 | 17CS72 | Advanced Computer Architectures | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 3 | 17CS73 | Machine Learning | CS/IS | 04 | | 03 | 60 | 40 | 100 | 4 |
| 4 | 17CS74x | Professional Elective 3 | CS/IS | 03 | | 03 | 60 | 40 | 100 | 3 |
| 5 | 17CS75x | Professional Elective 4 | CS/IS | 03 | | 03 | 60 | 40 | 100 | 3 |
| 6 | 17CSL76 | Machine Learning Laboratory | CS/IS | 01-Hour Instruction 02-Hour Practical | | 03 | 60 | 40 | 100 | 2 |
| 7 | 17CSL77 | Web Technology Laboratory with mini project | CS/IS | 01-Hour Instruction 02-Hour Practical | | 03 | 60 | 40 | 100 | 2 |
| 8 | 17CSP78 | Project Work Phase–I + Project work Seminar | CS/IS | | 03 | -- | -- | 100 | 100 | 2 |
| **TOTAL** | | | | **Theory:18 hours Practical and Project: 09 hours** | | **21** | **420** | **380** | **800** | **24** |

| Professional Elective-3 | | Professional Elective-4 | |
|---|---|---|---|
| 17CS741 | Natural Language Processing | 17CS751 | Soft and Evolutionary Computing |
| 17CS742 | Cloud Computing and its Applications | 17CS752 | Computer Vision and Robotics |
| 17CS743 | Information and Network Security | 17CS753 | Digital Image Processing |
| 17CS744 | Unix System Programming | 17CS754 | Storage Area Networks |

**1. Project Phase – I and Project Seminar:** Comprises of Literature Survey, Problem identification, Objectives and Methodology. CIE marks shall be based on the report covering Literature Survey, Problem identification, Objectives and Methodology and Seminar presentation skill.



H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI**
**Scheme of Teaching and Examination 2017-2018**
**Choice Based Credit System (CBCS)**

**B.E: Computer Science and Engineering**

**VIII SEMESTER**

| Sl. No | Course Code | Title | Teaching Department | Teaching Hours /Week | | Examination | | | | Credits |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | Theory | Practical/ Drawing | Duration in hours | SEE Marks | CIE Marks | Total Marks | |
| 1 | 17CS81 | Internet of Things and Applications | CS/IS | 4 | - | 3 | 60 | 40 | 100 | 4 |
| 2 | 17CS82 | Big Data Analytics | CS/IS | 4 | - | 3 | 60 | 40 | 100 | 4 |
| 3 | 17CS83X | Professional Elective-5 | CS/IS | 3 | - | 3 | 60 | 40 | 100 | 3 |
| 4 | 17CS84 | Internship/ Professional Practice | CS/IS | Industry Oriented | | 3 | 50 | 50 | 100 | 2 |
| 5 | 17CSP85 | Project Work-II | CS/IS | - | 6 | 3 | 100 | 100 | 200 | 6 |
| 6 | 17CSS86 | Seminar | CS/IS | - | 4 | - | - | 100 | 100 | 1 |
| **TOTAL** | | | | **Theory: 11 hours Project and Seminar: 10 hours** | | **15** | **330** | **370** | **700** | **20** |

| **Professional Elective -5** | |
|---|---|
| 17CS831 | High Performance Computing |
| 17CS832 | User Interface Design |
| 17CS833 | Network management |
| 17CS834 | System Modeling and Simulation |

**1. Internship/ Professional Practice:** 4 Weeks internship to be completed between the (VI and VII semester vacation) and/or (VII and VIII semester vacation) period.

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI**
**CHOICE BASED CREDIT SYSTEM (CBCS)**
**SCHEME OF TEACHING AND EXAMINATION 2017-2018**

**B.E. Computer Science & Engineering**
**B.E. Information Science and Engineering**

**V SEMESTER OPEN ELECTIVES**

| **Open Elective 1** | |
|---|---|
| 17CS561 | Programming in JAVA |
| 17CS562 | Artificial Intelligence |
| 17CS563 | Embedded Systems |
| 17CS564 | Dot Net framework for application development; |
| 17CS565 | Cloud Computing |

**VI SEMESTER**

| **Open Elective 2** | |
|---|---|
| 17CS661 | Mobile Application Development |
| 17CS662 | Big Data Analytics |
| 17CS663 | Wireless Networks and Mobile computing |
| 17CS664 | Python Application Programming |
| 17CS665 | Service Oriented Architecture |
| 17CS666 | Multicore Architecture and Programming |



H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**ENGINEERING MATHEMATICS-III**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 -2018)**
**SEMESTER – III**

| | | | |
|---|---|---|---|
| **Subject Code** | **17MAT31** | **IA Marks** | **40** |
| **Number of Lecture Hours/Week** | **04** | **Exam Marks** | **60** |
| **Total Number of Lecture Hours** | **50** | **Exam Hours** | **03** |

**CREDITS – 04**

| **Module -1** | **Teaching Hours** |
|---|---|
| **Fourier Series:** Periodic functions, Dirichlet's condition, Fourier Series of periodic functions with period $2\pi$ and with arbitrary period 2c. Fourier series of even and odd functions. Half range Fourier Series, practical harmonic analysis-Illustrative examples from engineering field. | **10Hours** |
| **Module -2** | |
| **Fourier Transforms:** Infinite Fourier transforms, Fourier sine and cosine transforms. Inverse Fourier transform.<br>**Z-transform:** Difference equations, basic definition, z-transform-definition, Standard z-transforms, Damping rule, Shifting rule, Initial value and final value theorems (without proof) and problems, Inverse z-transform. Applications of z-transforms to solve difference equations. | **10 Hours** |
| **Module – 3** | |
| **Statistical Methods:** Review of measures of central tendency and dispersion. Correlation-Karl Pearson's coefficient of correlation-problems. Regression analysis- lines of regression (without proof) –problems<br>**Curve Fitting:** Curve fitting by the method of least squares- fitting of the curves of the form, $y = ax + b$, $y = ax^2 + bx + c$ and $y = ae^{bx}$.<br>**Numerical Methods:** Numerical solution of algebraic and transcendental equations by Regula- Falsi Method and Newton-Raphson method. | **10 Hours** |
| **Module-4** | |
| **Finite differences:** Forward and backward differences, Newton's forward and backward interpolation formulae. Divided differences- Newton's divided difference formula. Lagrange's interpolation formula and inverse interpolation formula (all formulae without proof)-Problems.<br>**Numerical integration:** Simpson's $(1/3)^{th}$ and $(3/8)^{th}$ rules, Weddle's rule (without proof ) – Problems. | **10 Hours** |
| **Module-5** | |
| Vector integration: Line integrals-definition and problems, surface and volume integrals-definition, Green's theorem in a plane, Stokes and Gauss-divergence theorem(without proof) and problems.<br>**Calculus of Variations:** Variation of function and Functional, variational problems. Euler's equation, Geodesics, hanging chain, problems. | **10 Hours** |
| **Course outcomes:** | |

After Studying this course, students will be able to

- Know the use of periodic signals and Fourier series to analyze circuits and system communications.
- Explain the general linear system theory for continuous-time signals and digital signal processing using the Fourier Transform and z-transform.
- Employ appropriate numerical methods to solve algebraic and transcendental equations.
- Apply Green's Theorem, Divergence Theorem and Stokes' theorem in various applications in the field of electro-magnetic and gravitational fields and fluid flow problems.
- Determine the extremals of functionals and solve the simple problems of the calculus of variations.

**Question paper pattern:**

The question paper will have ten questions.
There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

**Text Books:**

1. B. S. Grewal," Higher Engineering Mathematics", Khanna publishers, 42nd edition, 2013.
2. B.V. Ramana "Higher Engineering Mathematics" Tata McGraw-Hill, 2006.

**Reference Books:**

1. N. P. Bali and Manish Goyal, "A text book of Engineering mathematics" , Laxmi publications, latest edition.
2. Kreyszig, "Advanced Engineering Mathematics " - 9th edition, Wiley.
3. H. K Dass and Er. Rajnish Verma ,"Higher Engineering Mathematics", S. Chand, 1st ed.

**ANALOG AND DIGITAL ELECTRONICS**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 -2018)**
**SEMESTER - III**

| | | | |
|---|---|---|---|
| **Subject Code** | **17CS32** | **IA Marks** | **40** |
| **Number of Lecture Hours/Week** | **04** | **Exam Marks** | **60** |
| **Total Number of Lecture Hours** | **50** | **Exam Hours** | **03** |

**CREDITS – 04**

| **Module -1** | **Teaching Hours** |
|---|---|
| **Field Effect Transistors**: Junction Field Effect Transistors, MOSFETs, Differences between JFETs and MOSFETs, Biasing MOSFETs, FET Applications, CMOS Devices. Wave-Shaping Circuits: Integrated Circuit(IC) Multivibrators. **Introduction to Operational Amplifier**: Ideal v/s practical Opamp, Performance Parameters, **Operational Amplifier Application Circuits**:Peak Detector Circuit, Comparator, Active Filters, Non-Linear Amplifier, Relaxation Oscillator, Current-To-Voltage Converter, Voltage-To-Current Converter.<br>**Text book 1:- Ch5: 5.2, 5.3, 5.5, 5.8, 5.9, 5.1.Ch13: 13.10.Ch 16: 16.3, 16.4. Ch 17: 7.12, 17.14, 17.15, 17.18, 17.19, 17.20, 17.21.)** | **10 Hours** |
| **Module -2** | |
| **The Basic Gates**: Review of Basic Logic gates, Positive and Negative Logic, Introduction to HDL. **Combinational Logic Circuits:** Sum-of-Products Method, Truth Table to Karnaugh Map, Pairs Quads, and Octets, Karnaugh Simplifications, Don't-care Conditions, Product-of-sums Method, Product-of-sums simplifications, Simplification by Quine-McClusky Method, Hazards and Hazard covers, HDL Implementation Models.<br>**Text book 2:- Ch2: 2.4, 2.5. Ch3: 3.2 to 3.11.** | **10 Hours** |
| **Module – 3** | |
| **Data-Processing Circuits:** Multiplexers, Demultiplexers, 1-of-16 Decoder, BCD to Decimal Decoders, Seven Segment Decoders, Encoders, Exclusive-OR Gates, Parity Generators and Checkers, Magnitude Comparator, Programmable Array Logic, Programmable Logic Arrays, HDL Implementation of Data Processing Circuits. Arithmetic Building Blocks, Arithmetic Logic Unit **Flip- Flops:** RS Flip-Flops, Gated Flip-Flops, Edge-triggered RS FLIP-FLOP, Edge-triggered D FLIP-FLOPs, Edge-triggered JK FLIP-FLOPs.<br>**Text book 2:- Ch 4:- 4.1 to 4.9, 4.11, 4.12, 4.14.Ch6:-6.7, 6.10.Ch8:- 8.1 to 8.5.** | **10 Hours** |
| **Module-4** | |
| **Flip- Flops:** FLIP-FLOP Timing, JK Master-slave FLIP-FLOP, Switch Contact Bounce Circuits, Various Representation of FLIP-FLOPs, HDL Implementation of FLIP-FLOP. **Registers:** Types of Registers, Serial In - Serial Out, Serial In - Parallel out, Parallel In - Serial Out, Parallel In - Parallel Out, Universal Shift Register, Applications of Shift Registers, Register implementation in HDL. **Counters:** Asynchronous Counters, Decoding Gates, Synchronous Counters, Changing the Counter Modulus.<br>**(Text book 2:- Ch 8: 8.6, 8.8, 8.9, 8.10, 8.13. Ch 9: 9.1 to 9.8. Ch 10: 10.1 to 10.4)** | **10 Hours** |

| Module-5 | |
|---|---|
| **Counters:** Decade Counters, Presettable Counters, Counter Design as a Synthesis problem, A Digital Clock, Counter Design using HDL. **D/A Conversion and A/D Conversion:** Variable, Resistor Networks, Binary Ladders, D/A Converters, D/A Accuracy and Resolution, A/D Converter-Simultaneous Conversion, A/D Converter-Counter Method, Continuous A/D Conversion, A/D Techniques, Dual-slope A/D Conversion, A/D Accuracy and Resolution.<br>**Text book 2:- Ch 10: 10.5 to 10.9. Ch 12: 12.1 to 12.10** | **10 Hours** |
| **Course outcomes:** After Studying this course, students will be able to<br>• Explain the operation of JFETs and MOSFETs , Operational Amplifier circuits and their application<br>• Explain Combinational Logic, Simplification Techniques using Karnaugh Maps, Quine McClusky technique.<br>• Demonstrate Operation of Decoders, Encoders, Multiplexers, Adders and Subtractors, working of Latches, Flip-Flops, Designing Registers, Counters, A/D and D/A Converters<br>• Design of Counters, Registers and A/D & D/A converters | |
| **Question paper pattern:**<br>The question paper will have ten questions.<br>There will be 2 questions from each module.<br>Each question will have questions covering all the topics under a module.<br>The students will have to answer 5 full questions, selecting one full question from each module. | |
| **Text Books:**<br>1. Anil K Maini, Varsha Agarwal: Electronic Devices and Circuits, Wiley, 2012.<br>2. Donald P Leach, Albert Paul Malvino & Goutam Saha: Digital Principles and Applications, 8th Edition, Tata McGraw Hill, 2015 | |
| **Reference Books:**<br>1. Stephen Brown, Zvonko Vranesic: Fundamentals of Digital Logic Design with VHDL, 2nd Edition, Tata McGraw Hill, 2005.<br>2. R D Sudhaker Samuel: Illustrative Approach to Logic Design, Sanguine-Pearson, 2010.<br>3. M Morris Mano: Digital Logic and Computer Design, 10th Edition, Pearson, 2008. | |

H. O. D
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**DATA STRUCTURES AND APPLICATIONS**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 -2018)**
**SEMESTER - III**

| | | | |
|---|---|---|---|
| **Subject Code** | **17CS33** | **IA Marks** | **40** |
| **Number of Lecture Hours/Week** | **04** | **Exam Marks** | **60** |
| **Total Number of Lecture Hours** | **50** | **Exam Hours** | **03** |

**CREDITS - 04**

| **Module -1** | **Teaching Hours** |
|---|---|
| **Introduction:** Data Structures, Classifications (Primitive & Non Primitive), Data structure Operations, Review of Arrays, Structures, Self-Referential Structures, and Unions. Pointers and Dynamic Memory Allocation Functions. Representation of Linear Arrays in Memory, Dynamically allocated arrays, **Array Operations**: Traversing, inserting, deleting, searching, and sorting. Multidimensional Arrays, Polynomials and Sparse Matrices. **Strings:** Basic Terminology, Storing, Operations and Pattern Matching algorithms. Programming Examples.<br>**Text 1: Ch 1: 1.2, Ch2: 2.2 -2.7**<br>**Text 2: Ch 1: 1.1 -1.4, Ch 3: 3.1-3.3,3.5,3.7, Ch 4: 4.1-4.9,4.14**<br>**Ref 3: Ch 1: 1.4** | **10 Hours** |
| **Module -2** | |
| **Stacks and Queues**<br>**Stacks:** Definition, Stack Operations, Array Representation of Stacks, Stacks using Dynamic Arrays, Stack Applications: Polish notation, Infix to postfix conversion, evaluation of postfix expression, **Recursion** - Factorial, GCD, Fibonacci Sequence, Tower of Hanoi, Ackerman's function. **Queues:** Definition, Array Representation, Queue Operations, Circular Queues, Circular queues using Dynamic arrays, Dequeues, Priority Queues, A Mazing Problem. Multiple Stacks and Queues. Programming Examples.<br><br>**Text 1: Ch3: 3.1 -3.7**<br>**Text 2: Ch6: 6.1 -6.3, 6.5, 6.7-6.10, 6.12, 6.13** | **10 Hours** |
| **Module – 3** | |
| **Linked Lists:** Definition, Representation of linked lists in Memory, Memory allocation; Garbage Collection. Linked list operations: Traversing, Searching, Insertion, and Deletion. Doubly Linked lists, Circular linked lists, and header linked lists. Linked Stacks and Queues. Applications of Linked lists – Polynomials, Sparse matrix representation. Programming Examples<br>**Text 1: Ch4: 4.1 -4.8 except 4.6**<br>**Text 2: Ch5: 5.1 – 5.10** | **10 Hours** |

| | |
|---|---|
| **Module-4** | |
| **Trees**: Terminology, Binary Trees, Properties of Binary trees, Array and linked Representation of Binary Trees, Binary Tree Traversals - Inorder, postorder, preorder; Additional Binary tree operations. Threaded binary trees, Binary Search Trees – Definition, Insertion, Deletion, Traversal, Searching, Application of Trees-Evaluation of Expression, Programming Examples<br>**Text 1: Ch5: 5.1 –5.5, 5.7**<br>**Text 2: Ch7: 7.1 – 7.9** | **10 Hours** |
| **Module-5** | |
| **Graphs**: Definitions, Terminologies, Matrix and Adjacency List Representation Of Graphs, Elementary Graph operations, Traversal methods: Breadth First Search and Depth First Search. **Sorting and Searching**: Insertion Sort, Radix sort, Address Calculation Sort. **Hashing:** Hash Table organizations, Hashing Functions, Static and Dynamic Hashing. **Files and Their Organization:** Data Hierarchy, File Attributes, Text Files and Binary Files, Basic File Operations, File Organizations and Indexing<br>**Text 1: Ch6: 6.1 –6.2, Ch 7:7.2, Ch 8:8.1-8.3**<br>**Text 2: Ch8: 8.1 – 8.7, Ch 9:9.1-9.3,9.7,9.9**<br>**Reference 2: Ch 16: 16.1 - 16.7** | **10 Hours** |

**Course outcomes:** After studying this course, students will be able to:

- Explain different types of data structures, operations and algorithms
- Apply searching and sorting operations on files
- Make use of stack, Queue, Lists, Trees and Graphs in problem solving.
- Develop all data structures in a high-level language for problem solving.

**Question paper pattern:**

The question paper will have ten questions.
There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

**Text Books:**


1. Fundamentals of Data Structures in C - Ellis Horowitz and Sartaj Sahni, 2nd edition, Universities Press,2014
2. Data Structures - Seymour Lipschutz, Schaum's Outlines, Revised 1st edition, McGraw Hill, 2014


**Reference Books:**


1. Data Structures: A Pseudo-code approach with C –Gilberg & Forouzan, 2nd edition, Cengage Learning,2014
2. Data Structures using C, , Reema Thareja, 3rd edition Oxford press, 2012
3. An Introduction to Data Structures with Applications- Jean-Paul Tremblay & Paul G. Sorenson, 2nd Edition, McGraw Hill, 2013
4. Data Structures using C - A M Tenenbaum, PHI, 1989
5. Data Structures and Program Design in C - Robert Kruse, 2nd edition, PHI, 1996


H.O.D

Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## COMPUTER ORGANIZATION

**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 -2018)**
**SEMESTER - III**

| | | | |
|---|---|---|---|
| **Subject Code** | **17CS34** | **IA Marks** | **40** |
| **Number of Lecture Hours/Week** | **04** | **Exam Marks** | **60** |
| **Total Number of Lecture Hours** | **50** | **Exam Hours** | **03** |

**CREDITS – 04**

| **Module -1** | **Teaching Hours** |
|---|---|
| Basic Structure of Computers: Basic Operational Concepts, Bus Structures, Performance – Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement. Machine Instructions and Programs: Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes, Assembly Language, Basic Input and Output Operations, Stacks and Queues, Subroutines, Additional Instructions, Encoding of Machine Instructions | **10Hours** |
| **Module -2** | |
| Input/Output Organization: Accessing I/O Devices, Interrupts – Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Controlling Device Requests, Exceptions, Direct Memory Access, Buses Interface Circuits, Standard I/O Interfaces – PCI Bus, SCSI Bus, USB. | **10 Hours** |
| **Module – 3** | |
| Memory System: Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Cache Memories – Mapping Functions, Replacement Algorithms, Performance Considerations, Virtual Memories, Secondary Storage. | **10 Hours** |
| **Module-4** | |
| Arithmetic: Numbers, Arithmetic Operations and Characters, Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed Operand Multiplication, Fast Multiplication, Integer Division, Floating-point Numbers and Operations. | **10 Hours** |
| **Module-5** | |
| Basic Processing Unit: Some Fundamental Concepts, Execution of a Complete Instruction, Multiple Bus Organization, Hard-wired Control, Micro programmed Control. Pipelining, Embedded Systems and Large Computer Systems: Basic Concepts of pipelining, Examples of Embedded Systems, Processor chips for embedded applications, Simple Microcontroller, The structure of General-Purpose Multiprocessors. | **10 Hours** |

**Course outcomes:** After studying this course, students will be able to:

- Explain the basic organization of a computer system.
- Demonstrate functioning of different sub systems, such as processor, Input/output, and memory.
- Illustrate hardwired control and micro programmed control. pipelining, embedded and other computing systems.
- Build simple arithmetic and logical units.

| **Question paper pattern:** |
| --- |
| The question paper will have ten questions.<br>There will be 2 questions from each module.<br>Each question will have questions covering all the topics under a module.<br>The students will have to answer 5 full questions, selecting one full question from each module. |
| **Text Books:** |
| 1. Carl Hamacher, Zvonko Vranesic, Safwat Zaky: Computer Organization, 5th Edition, Tata McGraw Hill, 2002. (Listed topics only from Chapters 1, 2, 4, 5, 6, 7, 8, 9 and12) |
| **Reference Books:** |
| 1. William Stallings: Computer Organization & Architecture, 9th Edition, Pearson, 2015. |

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engineering & Technology
Mijar, MOODBIDRI - 574 225

## UNIX AND SHELL PROGRAMMING

**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 -2018)**
**SEMESTER – III**

| | | | |
|---|---|---|---|
| **Subject Code** | **17CS35** | **IA Marks** | **40** |
| **Number of Lecture Hours/Week** | **03** | **Exam Marks** | **60** |
| **Total Number of Lecture Hours** | **40** | **Exam Hours** | **03** |

**CREDITS – 03**

| **Module -1** | **Teaching Hours** |
|---|---|
| Introduction, Brief history. Unix Components/Architecture. Features of Unix. The UNIX Environment and UNIX Structure, Posix and Single Unix specification. The login prompt. General features of Unix commands/ command structure. Command arguments and options. Understanding of some basic commands such as echo, printf, ls, who, date, passwd, cal, Combining commands. Meaning of Internal and external commands. The type command: knowing the type of a command and locating it. The man command knowing more about Unix commands and using Unix online manual pages. The man with keyword option and whatis. The more command and using it with other commands. Knowing the user terminal, displaying its characteristics and setting characteristics. Managing the non-uniform behaviour of terminals and keyboards. The root login. Becoming the super user: su command. The /etc/passwd and /etc/shadow files. Commands to add, modify and delete users.<br><br>**Topics from chapter 2 , 3 and 15 of text book 1,chapter 1 from text book 2** | **08 Hours** |
| **Module -2** | |
| Unix files. Naming files. Basic file types/categories. Organization of files. Hidden files. Standard directories. Parent child relationship. The home directory and the HOME variable. Reaching required files- the PATH variable, manipulating the PATH, Relative and absolute pathnames. Directory commands – pwd, cd, mkdir, rmdir commands. The dot (.) and double dots (..) notations to represent present and parent directories and their usage in relative path names. File related commands – cat, mv, rm, cp, wc and od commands. File attributes and permissions and knowing them. The ls command with options. Changing file permissions: the relative and absolute permissions changing methods. Recursively changing file permissions. Directory permissions.<br><br>**Topics from chapters 4, 5 and 6 of text book 1** | **08 Hours** |
| **Module – 3** | |
| The vi editor. Basics. The .exrc file. Different ways of invoking and quitting vi. Different modes of vi. Input mode commands. Command mode commands. The ex mode commands. Illustrative examples Navigation commands. Repeat command. Pattern searching. The search and replace command. The set, map and abbr commands. Simple examples using these commands.<br><br>The shells interpretive cycle. Wild cards and file name generation. Removing the special meanings of wild cards. Three standard files and redirection. Connecting commands: Pipe. Splitting the output: tee. Command substitution. Basic and Extended regular expressions. The grep, egrep. Typical examples involving different regular expressions.<br>**Topics from chapters 7, 8 and 13 of text book 1. Topics from chapter 2 and 9 ,10 of text book 2** | **08 Hours** |

| **Module-4** | |
|---|---|
| Shell programming. Ordinary and environment variables. The .profile. Read and readonly commands. Command line arguments. exit and exit status of a command. Logical operators for conditional execution. The test command and its shortcut. The if, while, for and case control statements. The set and shift commands and handling positional parameters. The here ( << ) document and trap command. Simple shell program examples. File inodes and the inode structure. File links – hard and soft links. Filters. Head and tail commands. Cut and paste commands. The sort command and its usage with different options. The umask and default file permissions. Two special files /dev/null and /dev/tty.<br><br>**Topics from chapter 11, 12, 14 of text book 1,chapter 17 from text book2** | **08 Hours** |
| **Module-5** | |
| Meaning of a process. Mechanism of process creation. Parent and child process. The ps command with its options. Executing a command at a specified point of time: at command. Executing a command periodically: cron command and the crontab file.. Signals. The nice and nohup commands. Background processes. The bg and fg command. The kill command. The find command with illustrative example.<br>Structure of a perl script. Running a perl script. Variables and operators. String handling functions. Default variables - $_ and $. – representing the current line and current line number. The range operator. Chop() and chomp() functions. Lists and arrays. The @- variable. The splice operator, push(), pop(), split() and join(). File handles and handling file – using open(), close() and die () functions.. Associative arrays – keys and value functions. Overview of decision making loop control structures – the foreach. Regular expressions – simple and multiple search patterns. The match and substitute operators. Defining and using subroutines.<br><br>**Topics from chapter 9 and 19 of text book 1. Topics from chapter 11 of reference book 1** | **08 Hours** |

**Course outcomes:**

After studying this course, students will be able to:

- Explain UNIX system and use different commands.
- Compile Shell scripts for certain functions on different subsystems.
- Demonstrate use of editors and Perl script writing

**Question paper pattern:**

The question paper will have ten questions.
There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

**Text Books:**


1. Sumitabha Das., Unix Concepts and Applications., 4th Edition., Tata McGraw Hill
2. Behrouz A. Forouzan, Richard F. Gilberg : UNIX and Shell Programming- Cengage Learning – India Edition. 2009.


**Reference Books:**


1. M.G. Venkatesh Murthy: UNIX & Shell Programming, Pearson Education.
2. Richard Blum , Christine Bresnahan : Linux Command Line and Shell Scripting Bible, 2nd Edition , Wiley,2014.


H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**DISCRETE MATHEMATICAL STRUCTURES**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 -2018)**
**SEMESTER – III**

| | | | |
|---|---|---|---|
| **Subject Code** | **17CS36** | **IA Marks** | **40** |
| **Number of Lecture Hours/Week** | **04** | **Exam Marks** | **60** |
| **Total Number of Lecture Hours** | **50** | **Exam Hours** | **03** |

**CREDITS – 04**

| **Module -1** | **Teaching Hours** |
|---|---|
| **Fundamentals of Logic**: Basic Connectives and Truth Tables, Logic Equivalence – The Laws of Logic, Logical Implication – Rules of Inference. Fundamentals of Logic contd.: The Use of Quantifiers, Quantifiers, Definitions and the Proofs of Theorems, | **10Hours** |
| **Module -2** | |
| **Properties of the Integers**: Mathematical Induction, The Well Ordering Principle – Mathematical Induction, Recursive Definitions. Principles of Counting. **Fundamental Principles of Counting:** The Rules of Sum and Product, Permutations, Combinations – The Binomial Theorem, Combinations with Repetition,. | **10 Hours** |
| **Module – 3** | |
| **Relations and Functions**: Cartesian Products and Relations, Functions – Plain and One-to-One, Onto Functions. The Pigeon-hole Principle, Function Composition and Inverse Functions. Properties of Relations, Computer Recognition – Zero-One Matrices and Directed Graphs, Partial Orders – Hasse Diagrams, Equivalence Relations and Partitions. | **10 Hours** |
| **Module-4** | |
| **The Principle of Inclusion and Exclusion**: The Principle of Inclusion and Exclusion, Generalizations of the Principle, Derangements – Nothing is in its Right Place, Rook Polynomials. **Recurrence Relations:** First Order Linear Recurrence Relation, The Second Order Linear Homogeneous Recurrence Relation with Constant Coefficients, | **10 Hours** |
| **Module-5** | |
| **Introduction to Graph Theory**: Definitions and Examples, Sub graphs, Complements, and Graph Isomorphism, Vertex Degree, Euler Trails and Circuits , **Trees**: Definitions, Properties, and Examples, Routed Trees, Trees and Sorting, Weighted Trees and Prefix Codes | **10 Hours** |

**Course outcomes:** After studying this course, students will be able to:

- Make use of propositional and predicate logic in knowledge representation and truth verification.
- Demonstrate the application of discrete structures in different fields of computer science.
- Solve problems using recurrence relations and generating functions.
- Apply different mathematical proofs, techniques in proving theorems.
- Compare graphs, trees and their applications.

**Question paper pattern:**

The question paper will have ten questions.
There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

**Text Books:**

1. Ralph P. Grimaldi: Discrete and Combinatorial Mathematics, , 5th Edition, Pearson Education. 2004. (Chapter 3.1, 3.2, 3.3, 3.4, Appendix 3, Chapter 2, Chapter 4.1, 4.2, Chapter 5.1 to 5.6, Chapter 7.1 to 7.4, Chapter 16.1, 16.2, 16.3, 16.5 to 16.9, and Chapter 14.1, 14.2, 14.3).

**Reference Books:**


1. Basavaraj S Anami and Venakanna S Madalli: Discrete Mathematics – A Concept based approach, Universities Press, 2016
2. Kenneth H. Rosen: Discrete Mathematics and its Applications, 6th Edition, McGraw Hill, 2007.
3. Jayant Ganguly: A Treatise on Discrete Mathematical Structures, Sanguine-Pearson, 2010.
4. D.S. Malik and M.K. Sen: Discrete Mathematical Structures: Theory and Applications, Thomson, 2004.
5. Thomas Koshy: Discrete Mathematics with Applications, Elsevier, 2005, Reprint 2008.


H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## ANALOG AND DIGITAL ELECTRONICS LABORATORY

**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 -2018)**
**SEMESTER - III**

| | | | |
|---|---|---|---|
| **Laboratory Code** | **17CSL37** | **IA Marks** | **40** |
| **Number of Lecture Hours/Week** | **01I + 02P** | **Exam Marks** | **60** |
| **Total Number of Lecture Hours** | **40** | **Exam Hours** | **03** |

**CREDITS – 02**

**Descriptions (if any)**

***Any simulation package like MultiSim / P-spice /Equivalent software may be used.***
Faculty-in-charge should demonstrate and explain the required hardware components and their functional Block diagrams, timing diagrams etc. Students have to prepare a write-up on the same and include it in the Lab record and to be evaluated.

**Laboratory Session-1:** Write-upon analog components; functional block diagram, Pin diagram (if any), waveforms and description. The same information is also taught in theory class; this helps the students to understand better.

**Laboratory Session-2:** Write-upon Logic design components, pin diagram (if any), Timing diagrams, etc. The same information is also taught in theory class; this helps the students to understand better.

***Note: These* TWO Laboratory sessions** are used to fill the gap between theory classes and practical sessions. Both sessions are to be evaluated for 40 marks as lab experiments.

**Laboratory Experiments:**

1. a) Design and construct a Schmitt trigger using Op-Amp for given UTP and LTP values and demonstrate its working.
   b) Design and implement a Schmitt trigger using Op-Amp using a simulation package for two sets of UTP and LTP values and demonstrate its working.
2. a) Design and construct a rectangular waveform generator (Op-Amp relaxation oscillator) for given frequency and demonstrate its working.
   b) Design and implement a rectangular waveform generator (Op-Amp relaxation oscillator) using a simulation package and demonstrate the change in frequency when all resistor values are doubled.
3. Design and implement an Astable multivibrator circuit using 555 timer for a given frequency and duty cycle.

NOTE: hardware and software results need to be compared

4. Design and implement Half adder, Full Adder, Half Subtractor, Full Subtractor using basic gates.
5. a) Given a 4-variable logic expression, simplify it using Entered Variable Map and realize the simplified logic expression using 8:1 multiplexer IC.
   b) Design and develop the Verilog /VHDL code for an 8:1 multiplexer. Simulate and verify its working.

6. a) Design and implement code converter I)Binary to Gray (II) Gray to Binary Code using basic gates.

7. Design and verify the Truth Table of 3-bit Parity Generator and 4-bit Parity Checker using basic Logic Gates with an even parity bit.

8. a) Realize a J-K Master / Slave Flip-Flop using NAND gates and verify its truth table.

b) Design and develop the Verilog / VHDL code for D Flip-Flop with positive-edge triggering. Simulate and verify it's working.

9. a) Design and implement a mod-n (n<8) synchronous up counter using J-K Flip-Flop ICs and demonstrate its working.

b) Design and develop the Verilog / VHDL code for mod-8 up counter. Simulate and verify it's working.

10. Design and implement an asynchronous counter using decade counter IC to count up from 0 to n (n<=9) and demonstrate on 7-segment display (using IC- 7447).

11. Generate a Ramp output waveform using DAC0800 (Inputs are given to DAC through IC74393 dual 4-bit binary counter).

**Study experiment**

12. To study 4-bitALU using IC-74181.

**Course outcomes:**

On the completion of this laboratory course, the students will be able to:

- Demonstrate various Electronic Devices like Cathode ray Oscilloscope, Signal generators, Digital Trainer Kit, Multimeters and components like Resistors, Capacitors, Op amp and Integrated Circuit.
- Design and demonstrate various combinational logic circuits.
- Design and demonstrate various types of counters and Registers using Flip-flops
- Make use of simulation package to design circuits.
- Infer the working and implementation of ALU.

**Conduction of Practical Examination:**

1. All laboratory experiments (1 to 11 nos) are to be included for practical examination.
2. Students are allowed to pick one experiment from the lot.
3. Strictly follow the instructions as printed on the cover page of answer script.
4. Marks distribution:
   a) For questions having part a only- Procedure + Conduction + Viva:**15 + 70 +15 =100 Marks**
   b) For questions having part a and b
      Part a- Procedure + Conduction + Viva:**09 + 42 +09= 60 Marks**
      Part b- Procedure + Conduction + Viva:**06 + 28 +06= 40 Marks**
5. **Change of experiment is allowed only once and marks allotted to the procedure part to be made zero.**

H.O.D.

Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

# DATA STRUCTURES LABORATORY

[As per Choice Based Credit System (CBCS) Scheme

| (Effective from the academic year 2017 -2018) SEMESTER - III | | | |
|---|---|---|---|
| **Laboratory Code** | **17CSL38** | **IA Marks** | **40** |
| **Number of Lecture Hours/Week** | **01I + 02P** | **Exam Marks** | **60** |
| **Total Number of Lecture Hours** | **40** | **Exam Hours** | **03** |
| **CREDITS - 02** | | | |

**Descriptions (if any)**

**Implement all the experiments in C Language under Linux / Windows environment.**

**Laboratory Experiments:**

1. Design, Develop and Implement a menu driven Program in C for the following **Array** operations
   a. Creating an Array of **N** Integer Elements
   b. Display of Array Elements with Suitable Headings
   c. Inserting an Element (**ELEM**) at a given valid Position (**POS**)
   d. Deleting an Element at a given valid Position(**POS**)
   e. Exit.

   Support the program with functions for each of the above operations.

2. Design, Develop and Implement a Program in C for the following operationson **Strings**
   a. Read a main String (**STR**), a Pattern String (**PAT**) and a Replace String (**REP**)
   b. Perform Pattern Matching Operation: Find and Replace all occurrences of **PAT** in **STR** with **REP** if **PAT** exists in **STR**. Report suitable messages in case **PAT** does not exist in **STR**

   Support the program with functions for each of the above operations. Don't use Built-in functions.

3. Design, Develop and Implement a menu driven Program in C for the following operations on **STACK** of Integers (Array Implementation of Stack with maximum size **MAX**)
   a. ***Push*** an Element on to Stack
   b. ***Pop*** an Element from Stack
   c. Demonstrate how Stack can be used to check ***Palindrome***
   d. Demonstrate ***Overflow*** and ***Underflow*** situations on Stack
   e. Display the status of Stack
   f. Exit

   Support the program with appropriate functions for each of the above operations

4. Design, Develop and Implement a Program in C for converting an Infix Expression to Postfix Expression. Program should support for both parenthesized and free parenthesized expressions with the operators: +, -, *, /, **%(Remainder), ^(Power)** and **alphanumeric** operands.

5. Design, Develop and Implement a Program in C for the following Stack Applications
   a. Evaluation of **Suffix expression** with single digit operands and operators: +, -, *, /, %, ^
   b. Solving **Tower of Hanoi** problem with **n** disks

6. Design, Develop and Implement a menu driven Program in C for the following operations on **Circular QUEUE** of Characters (Array Implementation of Queue with maximum size **MAX**)
    a. Insert an Element on to Circular QUEUE
    b. Delete an Element from Circular QUEUE
    c. Demonstrate ***Overflow*** and ***Underflow*** situations on Circular QUEUE
    d. Display the status of Circular QUEUE
    e. Exit

   Support the program with appropriate functions for each of the above operations

7. Design, Develop and Implement a menu driven Program in C for the following operations on **Singly Linked List (SLL)** of Student Data with the fields: ***USN, Name, Branch, Sem, PhNo***
    a. Create a **SLL** of **N** Students Data by using ***front insertion***.
    b. Display the status of **SLL** and count the number of nodes in it
    c. Perform Insertion / Deletion at End of **SLL**
    d. Perform Insertion / Deletion at Front of **SLL(Demonstration of stack)**
    e. Exit

8. Design, Develop and Implement a menu driven Program in C for the following operations on **Doubly Linked List (DLL)** of Employee Data with the fields: ***SSN, Name, Dept, Designation, Sal, PhNo***
    a. Create a **DLL** of **N** Employees Data by using ***end insertion***.
    b. Display the status of **DLL** and count the number of nodes in it
    c. Perform Insertion and Deletion at End of **DLL**
    d. Perform Insertion and Deletion at Front of **DLL**
    e. Demonstrate how this **DLL** can be used as **Double Ended Queue**
    f. Exit

9. Design, Develop and Implement a Program in C for the following operationson **Singly Circular Linked List (SCLL)** with header nodes
    a. Represent and Evaluate a Polynomial $\mathbf{P(x,y,z) = 6x^2y^2z-4yz^5+3x^3yz+2xy^5z-2xyz^3}$
    b. Find the sum of two polynomials **POLY1(x,y,z)** and **POLY2(x,y,z)** and store the result in **POLYSUM(x,y,z)**

   Support the program with appropriate functions for each of the above operations

10. Design, Develop and Implement a menu driven Program in C for the following operations on **Binary Search Tree (BST)** of Integers
    a. Create a BST of **N** Integers: 6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2
    b. Traverse the BST in Inorder, Preorder and Post Order
    c. Search the BST for a given element (**KEY**) and report the appropriate message
    e. Exit

11. Design, Develop and Implement a Program in C for the following operations on **Graph(G)** of Cities
    a. Create a Graph of **N** cities using Adjacency Matrix.
    b. Print all the nodes **reachable** from a given starting node in a digraph using DFS/**BFS** method

12. Given a File of **N** employee records with a set **K** of Keys(4-digit) which uniquely determine the records in file **F**. Assume that file **F** is maintained in memory by a Hash Table(HT) of **m** memory locations with **L** as the set of memory addresses (2-digit) of locations in HT. Let the keys in **K** and addresses in **L** are Integers. Design and develop a Program in C that uses Hash function **H: K →L** as H(**K**)=**K** mod **m (remainder** method), and implement hashing technique to map a given key **K** to the address space **L**. Resolve the collision (if any) using **linear probing**.

**Course outcomes:**

On the completion of this laboratory course, the students will be able to:

- Analyze and Compare various linear and non-linear data structures
- Demonstrate the working nature of different types of data structures and their applications
- Develop, analyze and evaluate the searching and sorting algorithms
- Choose the appropriate data structure for solving real world problems

**Conduction of Practical Examination:**

1. All laboratory experiments (**TWELVE** nos) are to be included for practical examination.
2. Students are allowed to pick one experiment from the lot.
3. Strictly follow the instructions as printed on the cover page of answer script
4. Marks distribution: Procedure + Conduction + Viva:**15 + 70 +15 (100)**
5. **Change of experiment is allowed only once and marks allotted to the procedure part to be made zero.**

H.O.D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## ENGINEERING MATHEMATICS-IV

**[As per Choice Based Credit System (CBCS) scheme]**

**(Effective from the academic year 2017 -2018)**

**SEMESTER – IV**

| | | | |
|---|---|---|---|
| Subject Code | 17MAT41 | IA Marks | 40 |
| Number of Lecture Hours/Week | 04 | Exam Marks | 60 |
| Total Number of Lecture Hours | 50 | Exam Hours | 03 |

**CREDITS – 04**

| **Module 1** | **Teaching Hours** |
|---|---|
| **Numerical Methods:** Numerical solution of ordinary differential equations of first order and first degree, Taylor's series method, modified Euler's method. Runge - Kutta method of fourth order, Milne's and Adams-Bashforth predictor and corrector methods (No derivations of formulae-single step computation only). | **10 Hours** |
| **Module 2** | |
| **Numerical Methods:** Numerical solution of second order ordinary differential equations, Runge-Kutta method and Milne's method. (No derivations of formulae-single step computation only).<br>**Special Functions:** Series solution of Bessel's differential equation leading to $J_n(x)$-Bessel's function of first kind. Basic properties and orthogonality. Series solution of Legendre's differential equation leading to $P_n(x)$-Legendre polynomials. Rodrigue's formula, problems | **10 Hours** |
| **Module 3** | |
| **Complex Variables:** Review of a function of a complex variable, limits, continuity, differentiability. Analytic functions-Cauchy-Riemann equations in cartesian and polar forms. Properties and construction of analytic functions. Complex line integrals-Cauchy's theorem and Cauchy's integral formula, Residue, poles, Cauchy's Residue theorem ( without proof) and problems.<br>**Transformations:** Conformal transformations-Discussion of transformations: $w = z^2$, $w = e^z$, $w = z + (1/z)$ $(z \neq 0)$, Bilinear transformations-problems. | **10 Hours** |
| **Module 4** | |
| **Probability Distributions:** Random variables (discrete and continuous), probability functions. Poisson distributions, geometric distribution, uniform distribution, exponential and normal distributions, Problems. **Joint probability distribution:** Joint Probability distribution for two variables, expectation, covariance, correlation coefficient. | **10 Hours** |
| **Module 5** | |
| **Sampling Theory:** Sampling, Sampling distributions, standard error, test of hypothesis for means and proportions, confidence limits for means, student's t-distribution, Chi-square distribution as a test of goodness of fit. **Stochastic process:** Stochastic process, probability vector, stochastic matrices, fixed points, regular stochastic matrices, Markov chains, higher transition probability. | **10 Hours** |
| **Course Outcomes:** After studying this course, students will be able to: | |
| • Solve first and second order ordinary differential equation arising in flow problems using single step and multistep numerical methods.<br>• Illustrate problems of potential theory, quantum mechanics and heat conduction by employing notions and properties of Bessel's functions and Legendre's polynomials.<br>• Explain the concepts of analytic functions, residues, poles of complex potentials and describe | |

| |
|---|
| conformal and Bilinear transformation arising in field theory and signal processing.<br>• Develop probability distribution of discrete, continuous random variables and joint probability distribution occurring in digital signal processing, information theory and design engineering.<br>• Demonstrate testing of hypothesis of sampling distributions and illustrate examples of Markov chains related to discrete parameter stochastic process. |
| **Question paper pattern:** |
| The question paper will have ten questions.<br>There will be 2 questions from each module.<br>Each question will have questions covering all the topics under a module.<br>The students will have to answer 5 full questions, selecting one full question from each module. |
| **Text Books:** |
| 1. B.V.Ramana "Higher Engineering Mathematics" Tata McGraw-Hill, 2006.<br>2. B. S. Grewal," Higher Engineering Mathematics", Khanna publishers, 42nd edition, 2013. |
| **Reference Books:** |
| 1. N P Bali and Manish Goyal, "A text book of Engineering mathematics" , Laxmi publications, latest edition.<br>2. Kreyszig, "Advanced Engineering Mathematics " - 9th edition, Wiley, 2013.<br>3. H. K Dass and Er. RajnishVerma, "Higher Engineering Mathematics", S. Chand, 1st ed, 2011. |

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**OBJECT ORIENTED CONCEPTS**

**[As per Choice Based Credit System (CBCS) scheme]**

**(Effective from the academic year 2017 -2018)**

**SEMESTER – IV**

| | | | |
|---|---|---|---|
| Subject Code | 17CS42 | IA Marks | 40 |
| Number of Lecture Hours/Week | 03 | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 03**

| **Module 1** | **Teaching Hours** |
|---|---|
| **Introduction to Object Oriented Concepts:** A Review of structures, Procedure–Oriented Programming system, Object Oriented Programming System, Comparison of Object Oriented Language with C, Console I/O, variables and reference variables, Function Prototyping, Function Overloading. **Class and Objects:** Introduction, member functions and data, objects and functions, objects and arrays, Namespaces, Nested classes, Constructors, Destructors. **Text book 1: Ch 1: 1.1 to 1.9 Ch 2: 2.1 to 2.6 Ch 4: 4.1 to 4.2** | **08 Hours** |
| **Module 2** | |
| **Introduction to Java**: Java's magic: the Byte code; Java Development Kit (JDK); the Java Buzzwords, Object-oriented programming; Simple Java programs. Data types, variables and arrays, Operators, Control Statements. **Text book 2: Ch:1 Ch: 2 Ch:3 Ch:4 Ch:5** | **08 Hours** |
| **Module 3** | |
| **Classes, Inheritance, Exceptions, Packages and Interfaces:** Classes: Classes fundamentals; Declaring objects; Constructors, this keyword, garbage collection. **Inheritance:** inheritance basics, using super, creating multi level hierarchy, method overriding. **Exception handling:** Exception handling in Java. Packages, Access Protection, Importing Packages, Interfaces. **Text book 2: Ch:6 Ch: 8 Ch:9 Ch:10** | **08 Hours** |
| **Module 4** | |
| **Multi Threaded Programming, Event Handling:** Multi Threaded Programming: What are threads? How to make the classes threadable ; Extending threads; Implementing runnable; Synchronization; Changing state of the thread; Bounded buffer problems, read-write problem, producer consumer problems. **Event Handling:** Two event handling mechanisms; The delegation event model; Event classes; Sources of events; Event listener interfaces; Using the delegation event model; Adapter classes; Inner classes. **Text book 2: Ch 11: Ch: 22** | **08 Hours** |
| **Module 5** | |
| **The Applet Class:** Introduction, Two types of Applets; Applet basics; Applet Architecture; An Applet skeleton; Simple Applet display methods; Requesting repainting; Using the Status Window; The HTML APPLET tag; Passing parameters to Applets; getDocumentbase() and getCodebase(); ApletContext and showDocument(); The AudioClip Interface; The AppletStub Interface;Output to the Console. **Swings:** Swings: The origins of Swing; Two key Swing features; Components and Containers; The Swing Packages; A simple Swing Application; Create a Swing Applet; Jlabel and ImageIcon; JTextField;The Swing Buttons; JTabbedpane; JScrollPane; JList; JComboBox; JTable. **Text book 2: Ch 21: Ch: 29 Ch: 30** | **08 Hours** |

**Course Outcomes:** After studying this course, students will be able to

- Explain the object-oriented concepts and JAVA.
- Develop computer programs to solve real world problems in Java.
- Develop simple GUI int erfaces for a computer program to interact with users, and to **comprehend** the event-based GUI handling principles using Applets and swings.

**Question paper pattern:**

The question paper will have ten questions.
There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

**Text Books:**

1. Sourav Sahay, Object Oriented Programming with C++ , 2nd Ed, Oxford University Press,2006
   (Chapters 1, 2, 4)
2. Herbert Schildt, Java The Complete Reference, 7th Edition, Tata McGraw Hill, 2007.
   (Chapters 1, 2, 3, 4, 5, 6, 8, 9,10, 11, 21, 22, 29, 30)

**Reference Book:**

1. Mahesh Bhave and Sunil Patekar, "Programming with Java", First Edition, Pearson Education,2008, ISBN:9788131720806
2. Herbert Schildt, The Complete Reference C++, 4th Edition, Tata McGraw Hill, 2003.
3. Stanley B.Lippmann, Josee Lajore, C++ Primer, 4th Edition, Pearson Education, 2005.
4. Rajkumar Buyya,S Thamarasi selvi, xingchen chu, Object oriented Programming with java, Tata McGraw Hill education private limited.
5. Richard A Johnson, Introduction to Java Programming and OOAD, CENGAGE Learning.
6. E Balagurusamy, Programming with Java A primer, Tata McGraw Hill companies.

**Note: Every institute shall organize a bridge organize on C++ either in the vacation or in the beginning of even semester.**

H.O.D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## DESIGN AND ANALYSIS OF ALGORITHMS

**[As per Choice Based Credit System (CBCS) scheme]**

**(Effective from the academic year 2017 -2018)**

**SEMESTER – IV**

| | | | |
|---|---|---|---|
| Subject Code | 17CS43 | IA Marks | 40 |
| Number of Lecture Hours/Week | 04 | Exam Marks | 60 |
| Total Number of Lecture Hours | 50 | Exam Hours | 03 |

**CREDITS – 04**

| **Module 1** | **Teaching Hours** |
|---|---|
| **Introduction:** What is an Algorithm? **(T2:1.1)**, Algorithm Specification **(T2:1.2)**, Analysis Framework **(T1:2.1)**, **Performance Analysis**: Space complexity, Time complexity **(T2:1.3)**. **Asymptotic Notations:** Big-Oh notation ($O$), Omega notation ($\Omega$), Theta notation ($\Theta$), and Little-oh notation ($o$), Mathematical analysis of Non-Recursive and recursive Algorithms with Examples **(T1:2.2, 2.3, 2.4)**. **Important Problem Types:** Sorting, Searching, String processing, Graph Problems, Combinatorial Problems. **Fundamental Data Structures:** Stacks, Queues, Graphs, Trees, Sets and Dictionaries. **(T1:1.3,1.4)** | **10 Hours** |
| **Module 2** | |
| **Divide and Conquer**: General method, Binary search, Recurrence equation for divide and conquer, Finding the maximum and minimum **(T2:3.1, 3.3, 3.4)**, Merge sort, Quick sort **(T1:4.1, 4.2)**, Strassen's matrix multiplication **(T2:3.8)**, Advantages and Disadvantages of divide and conquer. **Decrease and Conquer Approach:** Topological Sort. **(T1:5.3)** | **10 Hours** |
| **Module 3** | |
| **Greedy Method:** General method, Coin Change Problem, Knapsack Problem, Job sequencing with deadlines **(T2:4.1, 4.3, 4.5)**. **Minimum cost spanning trees:** Prim's Algorithm, Kruskal's Algorithm **(T1:9.1, 9.2)**. **Single source shortest paths:** Dijkstra's Algorithm **(T1:9.3)**. **Optimal Tree problem:** Huffman Trees and Codes **(T1:9.4)**. **Transform and Conquer Approach:** Heaps and Heap Sort **(T1:6.4)**. | **10 Hours** |
| **Module 4** | |
| **Dynamic Programming:** General method with Examples, Multistage Graphs **(T2:5.1, 5.2)**. **Transitive Closure:** Warshall's Algorithm, **All Pairs Shortest Paths:** Floyd's Algorithm, Optimal Binary Search Trees, Knapsack problem **((T1:8.2, 8.3, 8.4)**, Bellman-Ford Algorithm **(T2:5.4)**, Travelling Sales Person problem **(T2:5.9)**, Reliability design **(T2:5.8)**. | **10 Hours** |
| **Module 5** | |
| **Backtracking:** General method **(T2:7.1)**, N-Queens problem **(T1:12.1)**, Sum of subsets problem **(T1:12.1)**, Graph coloring **(T2:7.4)**, Hamiltonian cycles **(T2:7.5)**. **Branch and Bound:** Assignment Problem, Travelling Sales Person problem **(T1:12.2)**, **0/1 Knapsack problem (T2:8.2, T1:12.2):** LC Branch and Bound solution **(T2:8.2)**, FIFO Branch and Bound solution **(T2:8.2)**. **NP-Complete and NP-Hard problems:** Basic concepts, non-deterministic algorithms, P, NP, NP-Complete, and NP-Hard classes **(T2:11.1)**. | **10 Hours** |
| **Course Outcomes:** After studying this course, students will be able to | |
| • Describe computational solution to well known problems like searching, sorting etc.<br>• Estimate the computational complexity of different algorithms. | |

| |
|---|
| • Develop an algorithm using appropriate design strategies for problem solving. |
| **Question paper pattern:** |
| The question paper will have ten questions.<br>There will be 2 questions from each module.<br>Each question will have questions covering all the topics under a module.<br>The students will have to answer 5 full questions, selecting one full question from each module. |
| **Text Books:** |
| T1. Introduction to the Design and Analysis of Algorithms, Anany Levitin:, 2rd Edition, 2009. Pearson.<br>T2. Computer Algorithms/C++, Ellis Horowitz, Satraj Sahni and Rajasekaran, 2nd Edition, 2014, Universities Press |
| **Reference Books:** |
| 1. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 3rd Edition, PHI<br>2. Design and Analysis of Algorithms , S. Sridhar, Oxford (Higher Education) |

H.O.D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## MICROPROCESSORS AND MICROCONTROLLERS

**[As per Choice Based Credit System (CBCS) scheme]**

**(Effective from the academic year 2017 -2018)**

**SEMESTER – IV**

| | | | |
|---|---|---|---|
| Subject Code | 17CS44 | IA Marks | 40 |
| Number of Lecture Hours/Week | 04 | Exam Marks | 60 |
| Total Number of Lecture Hours | 50 | Exam Hours | 03 |

**CREDITS – 04**

| **Module 1** | **Teaching Hours** |
|---|---|
| **The x86 microprocessor:** Brief history of the x86 family, Inside the 8088/86, Introduction to assembly programming, Introduction to Program Segments, The Stack, Flag register, x86 Addressing Modes. **Assembly language programming:** Directives & a Sample Program, Assemble, Link & Run a program, More Sample programs, Control Transfer Instructions, Data Types and Data Definition, Full Segment Definition, Flowcharts and Pseudo code.<br>**Text book 1: Ch 1: 1.1 to 1.7, Ch 2: 2.1 to 2.7** | **10 Hours** |
| **Module 2** | |
| **x86:** Instructions sets description, **Arithmetic and logic instructions and programs:** Unsigned Addition and Subtraction, Unsigned Multiplication and Division, Logic Instructions, BCD and ASCII conversion, Rotate Instructions. **INT 21H and INT 10H Programming :** Bios INT 10H Programming , DOS Interrupt 21H. 8088/86 Interrupts, x86 PC and Interrupt Assignment.<br>**Text book 1: Ch 3: 3.1 to 3.5, Ch 4: 4.1 , 4.2 Chapter 14: 14.1 and 14.2** | **10 Hours** |
| **Module 3** | |
| **Signed Numbers and Strings:** Signed number Arithmetic Operations, String operations. **Memory and Memory interfacing:** Memory address decoding, data integrity in RAM and ROM, 16-bit memory interfacing. **8255 I/O programming:** I/O addresses MAP of x86 PC's, programming and interfacing the 8255.<br>**Text book 1: Ch 6: 6.1, 6.2. Ch 10: 10.2, 10.4, 10.5. Ch 11: 11.1 to 11.4** | **10 Hours** |
| **Module 4** | |
| Microprocessors versus Microcontrollers, **ARM Embedded Systems :**The RISC design philosophy, The ARM Design Philosophy, Embedded System Hardware, Embedded System Software, **ARM Processor Fundamentals :** Registers , Current Program Status Register , Pipeline, Exceptions, Interrupts, and the Vector Table , Core Extensions<br>**Text book 2:Ch 1:1.1 to 1.4, Ch 2:2.1 to 2.5** | **10 Hours** |
| **Module 5** | |
| **Introduction to the ARM Instruction Set :** Data Processing Instructions , Branch Instructions, Software Interrupt Instructions, Program Status Register Instructions, Coprocessor Instructions, Loading Constants, Simple programming exercises.<br>**Text book 2: Ch 3:3.1 to 3.6 ( Excluding 3.5.2)** | **10 Hours** |

**Course Outcomes:** After studying this course, students will be able to

- Differentiate between microprocessors and microcontrollers
- Develop assembly language code to solve problems
- Explain interfacing of various devices to x86 family and ARM processor
- Demonstrate interrupt routines for interfacing devices

**Question paper pattern:**

| |
|---|
| The question paper will have ten questions.<br>There will be 2 questions from each module.<br>Each question will have questions covering all the topics under a module.<br>The students will have to answer 5 full questions, selecting one full question from each module. |
| **Text Books:** |
| 1. Muhammad Ali Mazidi, Janice Gillispie Mazidi, Danny Causey, The x86 PC Assembly Language Design and Interfacing, 5th Edition, Pearson, 2013.<br>2. **ARM system developers guide**, Andrew N Sloss, Dominic Symes and Chris Wright, Elsevier,Morgan Kaufman publishers, 2008. |
| **Reference Books:** |
| 1. Douglas V. Hall: Microprocessors and Interfacing, Revised 2nd Edition, TMH, 2006.<br>2. K. Udaya Kumar & B.S. Umashankar : Advanced Microprocessors & IBM-PC Assembly Language Programming, TMH 2003.<br>3. Ayala : The 8086 Microprocessor: programming and interfacing - 1st edition, Cengage Learning<br>4. The Definitive Guide to the ARM Cortex-M3, by Joseph Yiu, 2nd Edition , Newnes, 2009<br>5. The Insider's Guide to the ARM7 based microcontrollers, Hitex Ltd.,1st edition, 2005<br>6. ARM System-on-Chip Architecture, Steve Furber, Second Edition, Pearson, 2015<br>7. Architecture, Programming and Interfacing of Low power Processors- ARM7, Cortex-M and MSP430, Lyla B Das Cengage Learning, 1st Edition |

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**SOFTWARE ENGINEERING**

**[As per Choice Based Credit System (CBCS) scheme]**

**(Effective from the academic year 2017 -2018)**

**SEMESTER – IV**

| | | | |
|---|---|---|---|
| Subject Code | 17CS45 | IA Marks | 40 |
| Number of Lecture Hours/Week | 04 | Exam Marks | 60 |
| Total Number of Lecture Hours | 50 | Exam Hours | 03 |

**CREDITS – 04**

| | |
|---|---|
| **Module 1** | **Teaching Hours** |
| **Introduction**: Software Crisis, Need for Software Engineering. Professional Software Development, Software Engineering Ethics. Case Studies.<br>**Software Processes**: Models: Waterfall Model **(Sec 2.1.1)**, Incremental Model **(Sec 2.1.2)** and Spiral Model **(Sec 2.1.3)**. Process activities.<br>**Requirements Engineering**: Requirements Engineering Processes **(Chap 4)**. Requirements Elicitation and Analysis **(Sec 4.5)**. Functional and non-functional requirements **(Sec 4.1)**. The software Requirements Document **(Sec 4.2)**. Requirements Specification **(Sec 4.3)**. Requirements validation **(Sec 4.6)**. Requirements Management **(Sec 4.7)**. | **12 Hours** |
| **Module 2** | |
| **System Models**: Context models **(Sec 5.1)**. Interaction models **(Sec 5.2)**. Structural models **(Sec 5.3)**. Behavioral models **(Sec 5.4)**. Model-driven engineering **(Sec 5.5)**.<br>**Design and Implementation**: Introduction to RUP **(Sec 2.4)**, Design Principles **(Chap 17)**. Object-oriented design using the UML **(Sec 7.1)**. Design patterns **(Sec 7.2)**. Implementation issues **(Sec 7.3)**. Open source development **(Sec 7.4)**. | **11 Hours** |
| **Module 3** | |
| **Software Testing**: Development testing **(Sec 8.1)**, Test-driven development **(Sec 8.2)**, Release testing **(Sec 8.3)**, User testing **(Sec 8.4)**. Test Automation **(Page no 42, 70,212, 231,444,695)**.<br>**Software Evolution**: Evolution processes **(Sec 9.1)**. Program evolution dynamics **(Sec 9.2)**. Software maintenance **(Sec 9.3)**. Legacy system management **(Sec 9.4)**. | **9 Hours** |
| **Module 4** | |
| **Project Planning**: Software pricing **(Sec 23.1)**. Plan-driven development **(Sec 23.2)**. Project scheduling **(Sec 23.3)**: Estimation techniques **(Sec 23.5)**. **Quality management**: Software quality **(Sec 24.1)**. Reviews and inspections **(Sec 24.3)**. Software measurement and metrics **(Sec 24.4)**. Software standards **(Sec 24.2)** | **10 Hours** |
| **Module 5** | |
| **Agile Software Development:** Coping with Change **(Sec 2.3)**, The Agile Manifesto: Values and Principles. Agile methods: SCRUM (Ref "**The SCRUM Primer, Ver 2.0**") and Extreme Programming **(Sec 3.3)**. Plan-driven and agile development **(Sec 3.2)**. Agile project management **(Sec 3.4)**, Scaling agile methods **(Sec 3.5)**: | **8 Hours** |
| **Course Outcomes:** After studying this course, students will be able to: | |
| • Design a software system, component, or process to meet desired needs within realistic constraints.<br>• Assess professional and ethical responsibility<br>• Function on multi-disciplinary teams<br>• Make use of techniques, skills, and modern engineering tools necessary for engineering | |

practice

- Comprehend software systems or parts of software systems.

**Question paper pattern:**

The question paper will have ten questions.
There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

**Text Books:**

1. Ian Sommerville: Software Engineering, 9th Edition, Pearson Education, 2012.
(Listed topics only from Chapters 1,2,3,4, 5, 7, 8, 9, 23, and 24)
2. The SCRUM Primer, Ver 2.0, http://www.goodagile.com/scrumprimer/scrumprimer20.pdf

**Reference Books:**

1. Roger S. Pressman: Software Engineering-A Practitioners approach, 7th Edition, Tata McGraw Hill.
2. Pankaj Jalote: An Integrated Approach to Software Engineering, Wiley India

**Web Reference for eBooks on Agile:**

1. http://agilemanifesto.org/
2. http://www.jamesshore.com/Agile-Book/

H.O.D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## DATA COMMUNICATION

**[As per Choice Based Credit System (CBCS) scheme]**

**(Effective from the academic year 2017 -2018)**

**SEMESTER – IV**

| | | | |
|---|---|---|---|
| Subject Code | 17CS46 | IA Marks | 40 |
| Number of Lecture Hours/Week | 04 | Exam Marks | 60 |
| Total Number of Lecture Hours | 50 | Exam Hours | 03 |

**CREDITS – 04**

| **Contents** | **Teaching Hours** |
|---|---|
| **Module 1** | |
| **Introduction:** Data Communications, Networks, Network Types, Internet History, Standards and Administration, **Networks Models**: Protocol Layering, TCP/IP Protocol suite, The OSI model, **Introduction to Physical Layer-1:** Data and Signals, Digital Signals, Transmission Impairment, Data Rate limits, Performance, **Digital Transmission**: Digital to digital conversion (Only Line coding: Polar, Bipolar and Manchester coding). | **10 Hours** |
| **Module 2** | |
| **Physical Layer-2:** Analog to digital conversion (only PCM), Transmission Modes, **Analog Transmission**: Digital to analog conversion, **Bandwidth Utilization**: Multiplexing and Spread Spectrum, **Switching**: Introduction, Circuit Switched Networks and Packet switching. | **10 Hours** |
| **Module 3** | |
| **Error Detection and Correction**: Introduction, Block coding, Cyclic codes, Checksum, Forward error correction, **Data link control**: DLC services, Data link layer protocols, HDLC, and Point to Point protocol (Framing, Transition phases only). | **10 Hours** |
| **Module 4** | |
| **Media Access control**: Random Access, Controlled Access and Channelization, **Wired LANs Ethernet**: Ethernet Protocol, Standard Ethernet, Fast Ethernet, Gigabit Ethernet and 10 Gigabit Ethernet, **Wireless LANs:** Introduction, IEEE 802.11 Project and Bluetooth. | **10 Hours** |
| **Module 5** | |
| **Other wireless Networks**: WIMAX, Cellular Telephony, Satellite networks, **Network layer Protocols** : Internet Protocol, ICMPv4,Mobile IP, **Next generation IP**: IPv6 addressing, The IPv6 Protocol, The ICMPv6 Protocol and Transition from IPv4 to IPv6. | **10 Hours** |

**Course Outcomes:** After studying this course, students will be able to

- Illustrate basic computer network technology.
- Identify the different types of network topologies and protocols.
- List and explain the layers of the OSI model and TCP/IP model.
- Comprehend the different types of network devices and their functions within a network
- Demonstrate subnetting and routing mechanisms.

**Question paper pattern:**

The question paper will have ten questions.
There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

| Text Book: |
| --- |
| Behrouz A. Forouzan, Data Communications and Networking 5E, 5th Edition, Tata McGraw-Hill, 2013. (Chapters 1.1 to 1.5, 2.1 to 2.3, 3.1, 3.3 to 3.6, 4.1 to 4.3, 5.1, 6.1, 6.2, 8.1 to 8.3, 10.1 to 10.5, 11.1 to 11.4, 12.1 to 12.3, 13.1 to 13.5, 15.1 to 15.3, 16.1 to 16.3, 19.1 to 19.3, 22.1 to 22.4) |
| **Reference Books:** |
| 1. Alberto Leon-Garcia and Indra Widjaja: Communication Networks - Fundamental Concepts and Key architectures, 2nd Edition Tata McGraw-Hill, 2004.<br>2. William Stallings: Data and Computer Communication, 8th Edition, Pearson Education, 2007.<br>3. Larry L. Peterson and Bruce S. Davie: Computer Networks – A Systems Approach, 4th Edition, Elsevier, 2007.<br>4. Nader F. Mir: Computer and Communication Networks, Pearson Education, 2007 |

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## DESIGN AND ANALYSIS OF ALGORITHM LABORATORY

**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 -2018)**
**SEMESTER – IV**

| | | | |
|---|---|---|---|
| Subject Code | 17CSL47 | IA Marks | 40 |
| Number of Lecture Hours/Week | 01 I + 02 P | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 02**

**Description**

Design, develop, and implement the specified algorithms for the following problems using Java language under LINUX /Windows environment.Netbeans/Eclipse IDE tool can be used for development and demonstration.

**Experiments**

| | | |
|---|---|---|
| 1 | A | Create a Java class called ***Student***with the following details as variables within it.<br>(i) USN<br>(ii) Name<br>(iii) Branch<br>(iv) Phone<br>Write a Java program to create *nStudent* objects and print the USN, Name, Branch, and Phoneof these objects with suitable headings. |
| | B | Write a Java program to implement the Stack using arrays. Write Push(), Pop(), and Display() methods to demonstrate its working. |
| 2 | A | Design a superclass called ***Staff*** with details as StaffId, Name, Phone, Salary. Extend this class by writing three subclasses namely ***Teaching*** (domain, publications), ***Technical*** (skills), and ***Contract*** (period). Write a Java program to read and display at least 3 *staff* objects of all three categories. |
| | B | Write a Java class called ***Customer*** to store their name and date_of_birth. The date_of_birth format should be dd/mm/yyyy. Write methods to read customer data as <name, dd/mm/yyyy> and display as <name, dd, mm, yyyy> using StringTokenizer class considering the delimiter character as "/". |
| 3 | A | Write a Java program to read two integers *a* and*b*. Compute *a/b* and print, when *b* is not zero. Raise an exception when *b* is equal to zero. |
| | B | Write a Java program that implements a multi-thread application that has three threads. First thread generates a random integer for every 1 second; second thread computes the square of the number andprints; third thread will print the value of cube of the number. |
| 4 | | Sort a given set of ***n*** integer elements using **Quick Sort** method and compute its time complexity. Run the program for varied values of *n*> 5000 and record the time taken to sort. Plot a graph of the time taken versus ***n***on graph sheet. The elements can be read from a file or can be generated using the random number generator. Demonstrate using Java how the divide-and-conquer method works along with its time complexity analysis: worst case, average case and best case. |
| 5 | | Sort a given set of ***n*** integer elements using **Merge Sort** method and compute its time complexity. Run the program for varied values of *n*> 5000, and record the time taken to sort. Plot a graph of the time taken versus ***n***on graph sheet. The elements can be read from a file or can be generated using the random number generator. Demonstrate using Java how the divide- |

| | |
|---|---|
| | and-conquer method works along with its time complexity analysis: worst case, average case and best case. |
| 6 | Implement in Java, the **0/1 Knapsack** problem using (a) Dynamic Programming method (b) Greedy method. |
| 7 | From a given vertex in a weighted connected graph, find shortest paths to other vertices using **Dijkstra's algorithm**. Write the program in Java. |
| 8 | Find Minimum Cost Spanning Tree of a given connected undirected graph using **Kruskal'salgorithm.** Use Union-Find algorithms in your program. |
| 9 | Find Minimum Cost Spanning Tree of a given connected undirected graph using **Prim's algorithm.** |
| 10 | Write Java programs to<br>(a) Implement All-Pairs Shortest Paths problem using **Floyd's algorithm.**<br>(b) Implement **Travelling Sales Person problem** using Dynamic programming. |
| 11 | Design and implement in Java to find a **subset** of a given set S = {Sl, S2,.....,Sn} of ***n*** positive integers whose SUM is equal to a given positive integer ***d***. For example, if S ={1, 2, 5, 6, 8} and ***d***= 9, there are two solutions {1,2,6}and {1,8}. Display a suitable message, if the given problem instance doesn't have a solution. |
| 12 | Design and implement in Java to find all **Hamiltonian Cycles** in a connected undirected Graph G of ***n*** vertices using backtracking principle. |

**Course Outcomes:** The students should be able to:

- Design algorithms using appropriate design techniques (brute-force, greedy, dynamic programming, etc.)
- Develop variety of algorithms such as sorting, graph related, combinatorial, etc., in a high level language.
- Analyze and compare the performance of algorithms using language features.
- Apply and implement learned algorithm design techniques and data structuresto solve real-world problems.

**Conduction of Practical Examination:**

All laboratory experiments (Twelve problems) are to be included for practical examination. Students are allowed to pick one experiment from the lot.
To generate the data set use random number generator function.
Strictly follow the instructions as printed on the cover page of answer script for breakup of marks
**Marks distribution: Procedure + Conduction + Viva: 15 + 70 + 15 (100). Change of experiment is allowed only once and marks allotted to the procedure**

H.O.D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## MICROPROCESSOR AND MICROCONTROLLER LABORATORY

**[As per Choice Based Credit System (CBCS) scheme]**

**(Effective from the academic year 2017 -2018)**

**SEMESTER – IV**

| | | | |
|---|---|---|---|
| Subject Code | 17CSL48 | IA Marks | 40 |
| Number of Lecture Hours/Week | 01 I + 02 P | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 02**

**Description**

Demonstration and Explanation hardware components and Faculty in-charge should explain 8086 architecture, pin diagram in one slot. The second slot, the Faculty in-charge should explain instruction set types/category etc. Students have to prepare a write-up on the same and include it in the Lab record and to be evaluated.

Laboratory Session-1: Write-up on Microprocessors, 8086 Functional block diagram, Pin diagram and description. The same information is also taught in theory class; this helps the students to understand better.

Laboratory Session-2: Write-up on Instruction group, Timing diagrams, etc. The same information is also taught in theory class; this helps the students to understand better.

Note: These TWO Laboratory sessions are used to fill the gap between theory classes and practical sessions. Both sessions are evaluated as lab experiments for 20 marks.

**Experiments**

- Develop and execute the following programs using 8086 Assembly Language. Any suitable assembler like MASM/TASM/8086 kit or any equivalent software may be used.
- Program should have suitable comments.
- The board layout and the circuit diagram of the interface are to be provided to the student during the examination.
- Software Required: Open source ARM Development platform, KEIL IDE and Proteus for simulation

**SOFTWARE PROGRAMS: PART A**

1. Design and develop an assembly language program to search a key element "X" in a list of 'n' 16-bit numbers. Adopt Binary search algorithm in your program for searching.
2. Design and develop an assembly program to sort a given set of 'n' 16-bit numbers in ascending order. Adopt Bubble Sort algorithm to sort given elements.
3. Develop an assembly language program to reverse a given string and verify whether it is a palindrome or not. Display the appropriate message.
4. Develop an assembly language program to compute nCr using recursive procedure. Assume that 'n' and 'r' are non-negative integers.
5. Design and develop an assembly language program to read the current time and Date from the system and display it in the standard format on the screen.
6. To write and simulate ARM assembly language programs for data transfer, arithmetic and logical operations (Demonstrate with the help of a suitable program).
7. To write and simulate C Programs for ARM microprocessor using KEIL (Demonstrate with the help of a suitable program)

   **Note : To use KEIL one may refer the book: Insider's Guide to the ARM7 based microcontrollers, Hitex Ltd.,1st edition, 2005**

**HARDWARE PROGRAMS: PART B**

8. a. Design and develop an assembly program to demonstrate BCD Up-Down Counter (00-99) on the Logic Controller Interface.
   b. Design and develop an assembly program to read the status of two 8-bit inputs (X & Y) from the Logic Controller Interface and display X*Y.
9. Design and develop an assembly program to display messages "FIRE" and "HELP" alternately with flickering effects on a 7-segment display interface for a suitable period of time. Ensure a flashing rate that makes it easy to read both the messages (Examiner does not specify these delay values nor is it necessary for the student to compute these values).
10. Design and develop an assembly program to drive a Stepper Motor interface and rotate the motor in specified direction (clockwise or counter-clockwise) by N steps (Direction and N are specified by the examiner). Introduce suitable delay between successive steps. (Any arbitrary value for the delay may be assumed by the student).
11. Design and develop an assembly language program to
    a. Generate the Sine Wave using DAC interface (The output of the DAC is to be displayed on the CRO).
    b. Generate a Half Rectified Sine waveform using the DAC interface. (The output of the DAC is to be displayed on the CRO).
12. To interface LCD with ARM processor– ARM7TDMI/LPC2148. Write and execute programs in C language for displaying text messages and numbers on LCD
13. To interface Stepper motor with ARM processor– ARM7TDMI/LPC2148. Write a program to rotate stepper motor

**Study Experiments:**

1. Interfacing of temperature sensor with ARM freedom board (or any other ARM microprocessor board) and display temperature on LCD
2. To design ARM cortex based automatic number plate recognition system
3. To design ARM based power saving system

**Course Outcomes:** After studying this course, students will be able to

- Summarize 80x86 instruction sets and comprehend the knowledge of how assembly language works.
- Design and develop assembly programs using 80x86 assembly language instructions
- Infer functioning of hardware devices and interfacing them to x86 family
- Choose processors for various kinds of applications.

**Conduction of Practical Examination:**

- All laboratory experiments (all 7 + 6 nos) are to be included for practical examination.
- Students are allowed to pick one experiment from each of the lot.
- Strictly follow the instructions as printed on the cover page of answer script for breakup of marks
- PART –A: Procedure + Conduction + Viva: **08 + 35 +07 (50)**
- PART –B: Procedure + Conduction + Viva: **08 + 35 +07 (50)**
- Change of experiment is allowed only once and marks allotted to the procedure part to be made zero.

H.O.D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## MANAGEMENT AND ENTREPRENEURSHIP FOR IT INDUSTRY

**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017-2018)**
**SEMESTER – V**

| | | | |
|---|---|---|---|
| Subject Code | 17CS51 | IA Marks | 40 |
| Number of Lecture Hours/Week | 4 | Exam Marks | 60 |
| Total Number of Lecture Hours | 50 | Exam Hours | 03 |

**CREDITS – 04**

| **Module – 1** | **Teaching Hours** |
|---|---|
| **Introduction** - Meaning, nature and characteristics of management, scope and Functional areas of management, goals of management, levels of management, brief overview of evolution of management theories,. Planning- Nature, importance, types of plans, steps in planning, Organizing- nature and purpose, types of Organization, Staffing- meaning, process of recruitment and selection | **10 Hours** |
| **Module – 2** | |
| **Directing and controlling-** meaning and nature of directing, leadership styles, motivation Theories, Communication- Meaning and importance, Coordination- meaning and importance, Controlling- meaning, steps in controlling, methods of establishing control. | **10 Hours** |
| **Module – 3** | |
| **Entrepreneur** – meaning of entrepreneur, characteristics of entrepreneurs, classification and types of entrepreneurs, various stages in entrepreneurial process, role of entrepreneurs in economic development, entrepreneurship in India and barriers to entrepreneurship. Identification of business opportunities, market feasibility study, technical feasibility study, financial feasibility study and social feasibility study. | **10 Hours** |
| **Module – 4** | |
| **Preparation of project and ERP -** meaning of project, project identification, project selection, project report, need and significance of project report, contents, formulation, guidelines by planning commission for project report, **Enterprise Resource Planning: Meaning and Importance- ERP** and Functional areas of Management – Marketing / Sales- Supply Chain Management – Finance and Accounting – Human Resources – Types of reports and methods of report generation | **10 Hours** |
| **Module – 5** | |
| **Micro and Small Enterprises:** Definition of micro and small enterprises, characteristics and advantages of micro and small enterprises, steps in establishing micro and small enterprises, Government of India indusial policy 2007 on micro and small enterprises, case study (Microsoft), Case study(Captain G R Gopinath),case study (N R Narayana Murthy & Infosys), **Institutional support:** MSME-DI, NSIC, SIDBI, KIADB, KSSIDC, TECSOK, KSFC, DIC and District level single window agency, **Introduction to IPR.** | **10 Hours** |
| **Course outcomes:** The students should be able to: | |
| • Define management, organization, entrepreneur, planning, staffing, ERP and outline their importance in entrepreneurship<br>• Utilize the resources available effectively through ERP<br>• Make use of IPRs and institutional support in entrepreneurship | |
| **Question paper pattern:** | |

| |
|---|
| The question paper will have TEN questions.<br>There will be TWO questions from each module.<br>Each question will have questions covering all the topics under a module.<br>The students will have to answer FIVE full questions, selecting ONE full question from each module. |
| **Text Books:** |
| 1. Principles of Management -P. C. Tripathi, P. N. Reddy; Tata McGraw Hill, 4th / 6th Edition, 2010.<br>2. Dynamics of Entrepreneurial Development & Management -Vasant Desai Himalaya Publishing House.<br>3. Entrepreneurship Development -Small Business Enterprises -Poornima M Charantimath Pearson Education – 2006.<br>4. Management and Entrepreneurship - Kanishka Bedi- Oxford University Press-2017 |
| **Reference Books:** |
| 1. Management Fundamentals -Concepts, Application, Skill Development Robert Lusier – Thomson.<br>2. Entrepreneurship Development -S S Khanka -S Chand & Co.<br>3. Management -Stephen Robbins -Pearson Education /PHI -17th Edition, 2003 |

H.O.D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**COMPUTER NETWORKS**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017-2018)**
**SEMESTER – V**

| | | | |
|---|---|---|---|
| Subject Code | 17CS52 | IA Marks | 40 |
| Number of Lecture Hours/Week | 4 | Exam Marks | 60 |
| Total Number of Lecture Hours | 50 | Exam Hours | 03 |

**CREDITS – 04**

| **Module – 1** | **Teaching Hours** |
|---|---|
| **Application Layer:** Principles of Network Applications: Network Application Architectures, Processes Communicating, Transport Services Available to Applications, Transport Services Provided by the Internet, Application-Layer Protocols. The Web and HTTP: Overview of HTTP, Non-persistent and Persistent Connections, HTTP Message Format, User-Server Interaction: Cookies, Web Caching, The Conditional GET, File Transfer: FTP Commands & Replies, Electronic Mail in the Internet: SMTP, Comparison with HTTP, Mail Message Format, Mail Access Protocols, DNS; The Internet's Directory Service: Services Provided by DNS, Overview of How DNS Works, DNS Records and Messages, Peer-to-Peer Applications: P2P File Distribution, Distributed Hash Tables.<br>**T1: Chap 2** | **10 Hours** |
| **Module – 2** | |
| **Transport Layer :** Introduction and Transport-Layer Services: Relationship Between Transport and Network Layers, Overview of the Transport Layer in the Internet, Multiplexing and Demultiplexing: Connectionless Transport: UDP,UDP Segment Structure, UDP Checksum, Principles of Reliable Data Transfer: Building a Reliable Data Transfer Protocol, Pipelined Reliable Data Transfer Protocols, Go-Back-N, Selective repeat, Connection-Oriented Transport TCP: The TCP Connection, TCP Segment Structure, Round-Trip Time Estimation and Timeout, Reliable Data Transfer, Flow Control, TCP Connection Management, Principles of Congestion Control: The Causes and the Costs of Congestion, Approaches to Congestion Control.<br>**T1: Chap 3** | **10 Hours** |
| **Module – 3** | |
| **The Network layer**: What's Inside a Router?: Input Processing, Switching, Output Processing, Where Does Queuing Occur? Routing control plane, IPv6,A Brief foray into IP Security, Routing Algorithms: The Link-State (LS) Routing Algorithm, The Distance-Vector (DV) Routing Algorithm, Hierarchical Routing, Routing in the Internet, Intra-AS Routing in the Internet: RIP, Intra-AS Routing in the Internet: OSPF, Inter/AS Routing: BGP, Broadcast Routing Algorithms and Multicast.<br>**T1: Chap 4: 4.3-4.7** | **10 Hours** |
| **Module – 4** | |
| **Wireless and Mobile Networks**: Cellular Internet Access: An Overview of Cellular Network Architecture, 3G Cellular Data Networks: Extending the Internet to Cellular subscribers, On to 4G:LTE,Mobility management: Principles, | **10 Hours** |

| | |
|---|---|
| Addressing, Routing to a mobile node, Mobile IP, Managing mobility in cellular Networks, Routing calls to a Mobile user, Handoffs in GSM, Wireless and Mobility: Impact on Higher-layer protocols.<br>**T1: Chap: 6 : 6.4-6.8** | |
| **Module – 5** | |
| **Multimedia Networking:** Properties of video, properties of Audio, Types of multimedia Network Applications, Streaming stored video: UDP Streaming, HTTP Streaming, Adaptive streaming and DASH, content distribution Networks, case study: You Tube.<br>**Network Support for Multimedia:** Quality-of-Service (QoS) Guarantees: Resource Reservation and Call Admission<br>**T1: Chap: 7** | **10 Hours** |
| **Course outcomes:** The students should be able to: | |
| • Explain principles of application layer protocols<br>• Outline transport layer services and infer UDP and TCP protocols<br>• Classify routers, IP and Routing Algorithms in network layer<br>• Explain the Wireless and Mobile Networks covering IEEE 802.11 Standard<br>• Define Multimedia Networking and Network Management | |
| **Question paper pattern:**<br>The question paper will have TEN questions.<br>There will be TWO questions from each module.<br>Each question will have questions covering all the topics under a module.<br>The students will have to answer FIVE full questions, selecting ONE full question from each module. | |
| **Text Books:** | |
| 1. James F Kurose and Keith W Ross, Computer Networking, A Top-Down Approach, Sixth edition, Pearson,2017 . | |
| **Reference Books:** | |
| 1. Behrouz A Forouzan, Data and Communications and Networking, Fifth Edition, McGraw Hill, Indian Edition<br>2. Larry L Peterson and Brusce S Davie, Computer Networks, fifth edition, ELSEVIER<br>3. Andrew S Tanenbaum, Computer Networks, fifth edition, Pearson<br>4. Mayank Dave, Computer Networks, Second edition, Cengage Learning | |

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## DATABASE MANAGEMENT SYSTEM

**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017-2018)**

**SEMESTER – V**

| | | | |
|---|---|---|---|
| Subject Code | 17CS53 | IA Marks | 40 |
| Number of Lecture Hours/Week | 4 | Exam Marks | 60 |
| Total Number of Lecture Hours | 50 | Exam Hours | 03 |

**CREDITS – 04**

| **Module – 1** | **Teaching Hours** |
|---|---|
| **Introduction to Databases:** Introduction, Characteristics of database approach, Advantages of using the DBMS approach, History of database applications. **Overview of Database Languages and Architectures:** Data Models, Schemas, and Instances. Three schema architecture and data independence, database languages, and interfaces, The Database System environment. **Conceptual Data Modelling using Entities and Relationships:** Entity types, Entity sets, attributes, roles, and structural constraints, Weak entity types, ER diagrams, examples, Specialization and Generalization.<br>**Textbook 1:Ch 1.1 to 1.8, 2.1 to 2.6, 3.1 to 3.10** | **10 Hours** |
| **Module – 2** | |
| **Relational Model**: Relational Model Concepts, Relational Model Constraints and relational database schemas, Update operations, transactions, and dealing with constraint violations. **Relational Algebra:** Unary and Binary relational operations, additional relational operations (aggregate, grouping, etc.) Examples of Queries in relational algebra. **Mapping Conceptual Design into a Logical Design:** Relational Database Design using ER-to-Relational mapping. **SQL:** SQL data definition and data types, specifying constraints in SQL, retrieval queries in SQL, INSERT, DELETE, and UPDATE statements in SQL, Additional features of SQL.<br>**Textbook 1: Ch4.1 to 4.5, 5.1 to 5.3, 6.1 to 6.5, 8.1; Textbook 2: 3.5** | **10 Hours** |
| **Module – 3** | |
| **SQL : Advances Queries:** More complex SQL retrieval queries, Specifying constraints as assertions and action triggers, Views in SQL, Schema change statements in SQL. **Database Application Development:** Accessing databases from applications, An introduction to JDBC, JDBC classes and interfaces, SQLJ, Stored procedures, Case study: The internet Bookshop. **Internet Applications:** The three-Tier application architecture, The presentation layer, The Middle Tier<br>**Textbook 1: Ch7.1 to 7.4; Textbook 2: 6.1 to 6.6, 7.5 to 7.7.** | **10 Hours** |
| **Module – 4** | |
| **Normalization: Database Design Theory** – Introduction to Normalization using Functional and Multivalued Dependencies: Informal design guidelines for relation schema, Functional Dependencies, Normal Forms based on Primary Keys, Second and Third Normal Forms, Boyce-Codd Normal Form, Multivalued Dependency and Fourth Normal Form, Join Dependencies and Fifth Normal Form. **Normalization Algorithms:** Inference Rules, Equivalence, and Minimal Cover, Properties of Relational Decompositions, Algorithms for Relational Database Schema Design, Nulls, Dangling tuples, and alternate Relational Designs, Further discussion of Multivalued dependencies and 4NF, Other dependencies and Normal Forms | **10 Hours** |

| | |
|---|---|
| **Textbook 1: Ch14.1 to 14.7, 15.1 to 15.6** | |
| **Module – 5** | |
| **Transaction Processing:** Introduction to Transaction Processing, Transaction and System concepts, Desirable properties of Transactions, Characterizing schedules based on recoverability, Characterizing schedules based on Serializability, Transaction support in SQL. **Concurrency Control in Databases:** Two-phase locking techniques for Concurrency control, Concurrency control based on Timestamp ordering, Multiversion Concurrency control techniques, Validation Concurrency control techniques, Granularity of Data items and Multiple Granularity Locking. **Introduction to Database Recovery Protocols:** Recovery Concepts, NO-UNDO/REDO recovery based on Deferred update, Recovery techniques based on immediate update, Shadow paging, Database backup and recovery from catastrophic failures<br>**Textbook 1: 20.1 to 20.6, 21.1 to 21.7, 22.1 to 22.4, 22.7.** | **10 Hours** |
| **Course outcomes:** The students should be able to:<br>• Summarize the concepts of database objects; enforce integrity constraints on a database using RDBMS.<br>• Use Structured Query Language (SQL) for database manipulation.<br>• Design simple database systems<br>• Design code for some application to interact with databases. | |
| **Question paper pattern:**<br>The question paper will have TEN questions.<br>There will be TWO questions from each module.<br>Each question will have questions covering all the topics under a module.<br>The students will have to answer FIVE full questions, selecting ONE full question from each module. | |
| **Text Books:**<br>1. Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.<br>2. Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill | |
| **Reference Books:**<br>1. Silberschatz Korth and Sudharshan, Database System Concepts, 6th Edition, Mc-GrawHill, 2013.<br>2. Coronel, Morris, and Rob, Database Principles Fundamentals of Design, Implementation and Management, Cengage Learning 2012. | |

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**AUTOMATA THEORY AND COMPUTABILITY**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017-2018)**
**SEMESTER – V**

| | | | |
|---|---|---|---|
| Subject Code | 17CS54 | IA Marks | 40 |
| Number of Lecture Hours/Week | 4 | Exam Marks | 60 |
| Total Number of Lecture Hours | 50 | Exam Hours | 03 |

**CREDITS – 04**

| **Module – 1** | **Teaching Hours** |
|---|---|
| **Why study the Theory of Computation, Languages and Strings**: Strings, Languages. A Language Hierarchy, Computation, **Finite State Machines (FSM)**: Deterministic FSM, Regular languages, Designing FSM, Nondeterministic FSMs, From FSMs to Operational Systems, Simulators for FSMs, Minimizing FSMs, Canonical form of Regular languages, Finite State Transducers, Bidirectional Transducers.<br>**Textbook 1: Ch 1,2, 3,4, 5.1 to 5.10** | **10 Hours** |
| **Module – 2** | |
| Regular Expressions (RE): what is a RE?, Kleene's theorem, Applications of REs, Manipulating and Simplifying REs. Regular Grammars: Definition, Regular Grammars and Regular languages. Regular Languages (RL) and Non-regular Languages: How many RLs, To show that a language is regular, Closure properties of RLs, to show some languages are not RLs.<br>**Textbook 1: Ch 6, 7, 8: 6.1 to 6.4, 7.1, 7.2, 8.1 to 8.4** | **10 Hours** |
| **Module – 3** | |
| Context-Free Grammars(CFG): Introduction to Rewrite Systems and Grammars, CFGs and languages, designing CFGs, simplifying CFGs, proving that a Grammar is correct, Derivation and Parse trees, Ambiguity, Normal Forms. Pushdown Automata (PDA): Definition of non-deterministic PDA, Deterministic and Non-deterministic PDAs, Non-determinism and Halting, alternative equivalent definitions of a PDA, alternatives that are not equivalent to PDA.<br>**Textbook 1: Ch 11, 12: 11.1 to 11.8, 12.1, 12.2, 12,4, 12.5, 12.6** | **10 Hours** |
| **Module – 4** | |
| Context-Free and Non-Context-Free Languages: Where do the Context-Free Languages(CFL) fit, Showing a language is context-free, Pumping theorem for CFL, Important closure properties of CFLs, Deterministic CFLs. Algorithms and Decision Procedures for CFLs: Decidable questions, Un-decidable questions. Turing Machine: Turing machine model, Representation, Language acceptability by TM, design of TM, Techniques for TM construction.<br>**Textbook 1: Ch 13: 13.1 to 13.5, Ch 14: 14.1, 14.2, Textbook 2: Ch 9.1 to 9.6** | **10 Hours** |
| **Module – 5** | |
| Variants of Turing Machines (TM), The model of Linear Bounded automata: Decidability: Definition of an algorithm, decidability, decidable languages, Undecidable languages, halting problem of TM, Post correspondence problem. Complexity: Growth rate of functions, the classes of P and NP, Quantum Computation: quantum computers, Church-Turing thesis.<br>**Textbook 2: Ch 9.7 to 9.8, 10.1 to 10.7, 12.1, 12.2, 12.8, 12.8.1, 12.8.2** | **10 Hours** |
| **Course outcomes:** The students should be able to:<br>• Tell the core concepts in automata theory and Theory of Computation | |

- Explain how to translate between different models of Computation (e.g., Deterministic and Non-deterministic and Software models).
- Interpret Grammars and Automata (recognizers) for different language classes and become knowledgeable about restricted models of Computation (Regular, Context Free) and their relative powers.
- Develop skills in formal reasoning and reduction of a problem to a formal model, with an emphasis on semantic precision and conciseness.
- Classify a problem with respect to different models of Computation.

**Question paper pattern:**

The question paper will have TEN questions.
There will be TWO questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer FIVE full questions, selecting ONE full question from each module.

**Text Books:**


1. Elaine Rich, Automata, Computability and Complexity, 1st Edition, Pearson Education,2012/2013
2. K L P Mishra, N Chandrasekaran , 3rd Edition, Theory of Computer Science, PhI, 2012.


**Reference Books:**


1. John E Hopcroft, Rajeev Motwani, Jeffery D Ullman, Introduction to AutomataTheory, Languages, and Computation, 3rd Edition, Pearson Education, 2013
2. Michael Sipser : Introduction to the Theory of Computation, 3rd edition, Cengage learning,2013
3. John C Martin, Introduction to Languages and The Theory of Computation, 3rd Edition, Tata McGraw –Hill Publishing Company Limited, 2013
4. Peter Linz, "An Introduction to Formal Languages and Automata", 3rd Edition, Narosa Publishers, 1998
5. Basavaraj S. Anami, Karibasappa K G, Formal Languages and Automata theory, Wiley India, 2012
6. C K Nagpal, Formal Languages and Automata Theory, Oxford University press, 2012.


H.O.D
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## OBJECT ORIENTED MODELING AND DESIGN

**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017-2018)**
**SEMESTER – V**

| | | | |
|---|---|---|---|
| Subject Code | 17CS551 | IA Marks | 40 |
| Number of Lecture Hours/Week | 3 | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 03**

| **Module – 1** | **Teaching Hours** |
|---|---|
| **Introduction, Modelling Concepts and Class Modelling:** What is Object orientation? What is OO development? OO Themes; Evidence for usefulness of OO development; OO modelling history. Modelling as Design technique: Modelling; abstraction; The Three models. Class Modelling: Object and Class Concept; Link and associations concepts; Generalization and Inheritance; A sample class model; Navigation of class models; Advanced Class Modelling, Advanced object and class concepts; Association ends; N-ary associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived Data; Packages.<br>**Text Book-1: Ch 1, 2, 3 and 4** | **8 Hours** |
| **Module – 2** | |
| UseCase Modelling and Detailed Requirements: Overview; Detailed object-oriented Requirements definitions; System Processes-A use case/Scenario view; Identifying Input and outputs-The System sequence diagram; Identifying Object Behaviour-The state chart Diagram; Integrated Object-oriented Models.<br>**Text Book-2:Chapter- 6:Page 210 to 250** | **8 Hours** |
| **Module – 3** | |
| Process Overview, System Conception and Domain Analysis: Process Overview: Development stages; Development life Cycle; System Conception: Devising a system concept; elaborating a concept; preparing a problem statement. Domain Analysis: Overview of analysis; Domain Class model: Domain state model; Domain interaction model; Iterating the analysis.<br>**Text Book-1:Chapter- 10,11,and 12** | **8 Hours** |
| **Module – 4** | |
| Use case Realization :The Design Discipline within up iterations: Object Oriented Design-The Bridge between Requirements and Implementation; Design Classes and Design within Class Diagrams; Interaction Diagrams-Realizing Use Case and defining methods; Designing with Communication Diagrams; Updating the Design Class Diagram; Package Diagrams-Structuring the Major Components; Implementation Issues for Three-Layer Design.<br>**Text Book-2: Chapter 8: page 292 to 346** | **8 Hours** |
| **Module – 5** | |
| Design Patterns: Introduction; what is a design pattern?, Describing design patterns, the catalogue of design patterns, Organizing the catalogue, How design patterns solve design problems, how to select a design patterns, how to use a design pattern; Creational patterns: prototype and singleton (only); structural patterns adaptor and proxy (only).<br>**Text Book-3: Ch-1: 1.1, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8,Ch-3,Ch-4.** | **8 Hours** |
| **Course outcomes:** The students should be able to: | |

- Describe the concepts of object-oriented and basic class modelling.
- Draw class diagrams, sequence diagrams and interaction diagrams to solve problems.
- Choose and apply a befitting design pattern for the given problem.

**Question paper pattern:**

The question paper will have TEN questions.
There will be TWO questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer FIVE full questions, selecting ONE full question from each module.

**Text Books:**


1. Michael Blaha, James Rumbaugh: Object Oriented Modelling and Design with UML,2nd Edition, Pearson Education,2005
2. Satzinger, Jackson and Burd: Object-Oriented Analysis & Design with the Unified Process, Cengage Learning, 2005.
3. Erich Gamma, Richard Helm, Ralph Johnson and john Vlissides: Design Patterns – Elements of Reusable Object-Oriented Software, Pearson Education,2007.


**Reference Books:**


1. Grady Booch et. al.: Object-Oriented Analysis and Design with Applications,3rd Edition,Pearson Education,2007.
2. 2.Frank Buschmann, RegineMeunier, Hans Rohnert, Peter Sommerlad, Michel Stal: Pattern –Oriented Software Architecture. A system of patterns , Volume 1, John Wiley and Sons.2007.
3. 3. Booch, Jacobson, Rambaugh : Object-Oriented Analysis and Design with Applications, 3rd edition, pearson, Reprint 2013


H.O.D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**ADVANCED JAVA AND J2EE**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017-2018)**
**SEMESTER – V**

| | | | |
|---|---|---|---|
| Subject Code | 17CS553 | IA Marks | 40 |
| Number of Lecture Hours/Week | 3 | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 03**

| **Module – 1** | **Teaching Hours** |
|---|---|
| **Enumerations, Autoboxing and Annotations(metadata):** Enumerations, Enumeration fundamentals, the values() and valueOf() Methods, java enumerations are class types, enumerations Inherits Enum, example, type wrappers, Autoboxing, Autoboxing and Methods, Autoboxing/Unboxing occurs in Expressions, Autoboxing/Unboxing, Boolean and character values, Autoboxing/Unboxing helps prevent errors, A word of Warning. Annotations, Annotation basics, specifying retention policy, Obtaining Annotations at run time by use of reflection, Annotated element Interface, Using Default values, Marker Annotations, Single Member annotations, Built-In annotations. | **8 Hours** |
| **Module – 2** | |
| **The collections and Framework:** Collections Overview, Recent Changes to Collections, The Collection Interfaces, The Collection Classes, Accessing a collection Via an Iterator, Storing User Defined Classes in Collections, The Random Access Interface, Working With Maps, Comparators, The Collection Algorithms, Why Generic Collections?, The legacy Classes and Interfaces, Parting Thoughts on Collections. | **8 Hours** |
| **Module – 3** | |
| String Handling :The String Constructors, String Length, Special String Operations, String Literals, String Concatenation, String Concatenation with Other Data Types, String Conversion and toString( ) Character Extraction, charAt( ), getChars( ), getBytes( ) toCharArray(), String Comparison, equals( ) and equalsIgnoreCase( ), regionMatches( ) startsWith( ) and endsWith( ), equals( ) Versus == , compareTo( ) Searching Strings, Modifying a String, substring( ), concat( ), replace( ), trim( ), Data Conversion Using valueOf( ), Changing the Case of Characters Within a String, Additional String Methods, StringBuffer , StringBuffer Constructors, length( ) and capacity( ), ensureCapacity( ), setLength( ), charAt( ) and setCharAt( ), getChars( ),append( ), insert( ), reverse( ), delete( ) and deleteCharAt( ), replace( ), substring( ), Additional StringBuffer Methods, StringBuilder<br>**Text Book 1: Ch 15** | **8 Hours** |
| **Module – 4** | |
| Background; The Life Cycle of a Servlet; Using Tomcat for Servlet Development; A simple Servlet; The Servlet API; The Javax.servlet Package; Reading Servlet Parameter; The Javax.servlet.http package; Handling HTTP Requests and Responses; Using Cookies; Session Tracking. Java Server Pages (JSP): JSP, JSP Tags, Tomcat, Request String, User Sessions, Cookies, Session Objects | **8 Hours** |

| | |
|---|---|
| **Text Book 1: Ch 31 Text Book 2: Ch 11** | |
| **Module – 5** | |
| The Concept of JDBC; JDBC Driver Types; JDBC Packages; A Brief Overview of the JDBC process; Database Connection; Associating the JDBC/ODBC Bridge with the Database; Statement Objects; ResultSet; Transaction Processing; Metadata, Data types; Exceptions.<br>**Text Book 2: Ch 06** | **8 Hours** |

**Course outcomes:** The students should be able to:

- Interpret the need for advanced Java concepts like enumerations and collections in developing modular and efficient programs
- Build client-server applications and TCP/IP socket programs
- Illustrate database access and details for managing information using the JDBC API
- Describe how servlets fit into Java-based web application architecture
- Develop reusable software components using Java Beans

**Question paper pattern:**

The question paper will have TEN questions.
There will be TWO questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer FIVE full questions, selecting ONE full question from each module.

**Text Books:**


1. Herbert Schildt: JAVA the Complete Reference, 7th/9th Edition, Tata McGraw Hill, 2007.
2. Jim Keogh: J2EE-TheCompleteReference, McGraw Hill, 2007.


**Reference Books:**


1. Y. Daniel Liang: Introduction to JAVA Programming, 7thEdition, Pearson Education, 2007.
2. Stephanie Bodoff et al: The J2EE Tutorial, 2nd Edition, Pearson Education,2004.
3. Uttam K Roy, Advanced JAVA programming, Oxford University press, 2015.


H.O.D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**ARTIFICIAL INTELLIGENCE**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 -2018)**
**SEMESTER – V**

| | | | |
|---|---|---|---|
| Subject Code | 17CS562 | IA Marks | 40 |
| Number of Lecture Hours/Week | 3 | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 03**

| **Module – 1** | **Teaching Hours** |
|---|---|
| What is artificial intelligence?, Problems, Problem Spaces and search, Heuristic search technique<br>**TextBook1: Ch 1, 2 and 3** | **8 Hours** |
| **Module – 2** | |
| **Knowledge Representation Issues, Using** Predicate Logic, Representing knowledge using Rules,<br>**TextBoook1: Ch 4, 5 and 6.** | **8 Hours** |
| **Module – 3** | |
| Symbolic Reasoning under Uncertainty, Statistical reasoning, Weak Slot and Filter Structures.<br>**TextBoook1: Ch 7, 8 and 9.** | **8 Hours** |
| **Module – 4** | |
| Strong slot-and-filler structures, Game Playing.<br>**TextBoook1: Ch 10 and 12** | **8 Hours** |
| **Module – 5** | |
| Natural Language Processing, Learning, Expert Systems.<br>**TextBook1: Ch 15,17 and 20** | **8 Hours** |

**Course outcomes:** The students should be able to:

- Identify the AI based problems
- Apply techniques to solve the AI problems
- Define learning and explain various learning techniques
- Discuss expert systems

**Question paper pattern:**
The question paper will have TEN questions.
There will be TWO questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer FIVE full questions, selecting ONE full question from each module.

**Text Books:**

1. E. Rich , K. Knight & S. B. Nair - Artificial Intelligence, 3/e, McGraw Hill.

**Reference Books:**

1. Artificial Intelligence: A Modern Approach, Stuart Rusell, Peter Norving, Pearson Education 2nd Edition.
1. Dan W. Patterson, Introduction to Artificial Intelligence and Expert Systems – Prentice Hal of India.
2. G. Luger, "Artificial Intelligence: Structures and Strategies for complex problem

Solving", Fourth Edition, Pearson Education, 2002.
3. Artificial Intelligence and Expert Systems Development by D W Rolston-Mc Graw hill.
4. N.P. Padhy "Artificial Intelligence and Intelligent Systems" , Oxford University Press-2015

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**DOT NET FRAMEWORK FOR APPLICATION DEVELOPMENT**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 -2018)**
**SEMESTER – V**

| | | | |
|---|---|---|---|
| Subject Code | 17CS564 | IA Marks | 40 |
| Number of Lecture Hours/Week | 3 | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 03**

| **Module – 1** | **Teaching Hours** |
|---|---|
| **Introducing Microsoft Visual C# and Microsoft Visual Studio 2015:** Welcome to C#, Working with variables, operators and expressions, Writing methods and applying scope, Using decision statements, Using compound assignment and iteration statements, Managing errors and exceptions<br>**T1: Chapter 1 – Chapter 6** | **8 Hours** |
| **Module – 2** | |
| **Understanding the C# object model:** Creating and Managing classes and objects, Understanding values and references, Creating value types with enumerations and structures, Using arrays<br>**Textbook 1: Ch 7 to 10** | **8 Hours** |
| **Module – 3** | |
| Understanding parameter arrays, Working with inheritance, Creating interfaces and defining abstract classes, Using garbage collection and resource management<br>**Textbook 1: Ch 11 to 14** | **8 Hours** |
| **Module – 4** | |
| **Defining Extensible Types with C#:** Implementing properties to access fields, Using indexers, Introducing generics, Using collections<br>**Textbook 1: Ch 15 to 18** | **8 Hours** |
| **Module – 5** | |
| Enumerating Collections, Decoupling application logic and handling events, Querying in-memory data by using query expressions, Operator overloading<br>**Textbook 1: Ch 19 to 22** | **8 Hours** |

**Course outcomes:** The students should be able to:

- Build applications on Visual Studio .NET platform by understanding the syntax and semantics of C#
- Demonstrate Object Oriented Programming concepts in C# programming language
- Design custom interfaces for applications and leverage the available built-in interfaces in building complex applications.
- Illustrate the use of generics and collections in C#
- Compose queries to query in-memory data and define own operator behaviour

**Question paper pattern:**
The question paper will have TEN questions.
There will be TWO questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer FIVE full questions, selecting ONE full question from each module.

**Text Books:**

1. John Sharp, Microsoft Visual C# Step by Step, 8th Edition, PHI Learning Pvt. Ltd. 2016

**Reference Books:**


1. Christian Nagel, "C# 6 and .NET Core 1.0", 1st Edition, Wiley India Pvt Ltd, 2016. Andrew Stellman and Jennifer Greene, "Head First C#", 3rd Edition, O'Reilly Publications, 2013.
2. Mark Michaelis, "Essential C# 6.0", 5th Edition, Pearson Education India, 2016.
3. Andrew Troelsen, "Prof C# 5.0 and the .NET 4.5 Framework", 6th Edition, Apress and Dreamtech Press, 2012.


H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## COMPUTER NETWORK LABORATORY
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017-2018)**
**SEMESTER – V**

| | | | |
|---|---|---|---|
| Subject Code | 17CSL57 | IA Marks | 40 |
| Number of Lecture Hours/Week | 01I + 02P | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 02**

**Description (If any):**

For the experiments below modify the topology and parameters set for the experiment and take multiple rounds of reading and analyze the results available in log files. Plot necessary graphs and conclude. Use NS2/NS3.

**Lab Experiments:**

**PART A**

1. Implement three nodes point – to – point network with duplex links between them. Set the queue size, vary the bandwidth and find the number of packets dropped.
2. Implement transmission of ping messages/trace route over a network topology consisting of 6 nodes and find the number of packets dropped due to congestion.
3. Implement an Ethernet LAN using n nodes and set multiple traffic nodes and plot congestion window for different source / destination.
4. Implement simple ESS and with transmitting nodes in wire-less LAN by simulation and determine the performance with respect to transmission of packets.
5. Implement and study the performance of GSM on NS2/NS3 (Using MAC layer) or equivalent environment.
6. Implement and study the performance of CDMA on NS2/NS3 (Using stack called Call net) or equivalent environment.

**PART B**

**Implement the following in Java:**

7. Write a program for error detecting code using CRC-CCITT (16- bits).
8. Write a program to find the shortest path between vertices using bellman-ford algorithm.
9. Using TCP/IP sockets, write a client – server program to make the client send the file name and to make the server send back the contents of the requested file if present.
10. Write a program on datagram socket for client/server to display the messages on client side, typed at the server side.
11. Write a program for simple RSA algorithm to encrypt and decrypt the data.
12. Write a program for congestion control using leaky bucket algorithm.

**Study Experiment / Project:**

**NIL**

**Course outcomes:** The students should be able to:

- Analyze and Compare various networking protocols.
- Demonstrate the working of different concepts of networking.
- Implement and analyze networking protocols in NS2 / NS3

**Conduction of Practical Examination:**

1. All laboratory experiments are to be included for practical examination.
2. Students are allowed to pick one experiment from part A and part B with lot.
3. Strictly follow the instructions as printed on the cover page of answer script

4. Marks distribution: Procedure + Conduction + Viva: 100

**Part A: 8+35+7 =50**

**Part B: 8+35+7 =50**

5. Change of experiment is allowed only once and marks allotted to the procedure part to be made zero.

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**DBMS LABORATORY WITH MINI PROJECT**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017-2018)**
**SEMESTER – V**

| | | | |
|---|---|---|---|
| Subject Code | 17CSL58 | IA Marks | 40 |
| Number of Lecture Hours/Week | 01I + 02P | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 02**

**Description (If any):**

**PART-A: SQL Programming (Max. Exam Mks. 50)**

- Design, develop, and implement the specified queries for the following problems using Oracle, MySQL, MS SQL Server, or any other DBMS under LINUX/Windows environment.
- Create Schema and insert at least 5 records for each table. Add appropriate database constraints.

**PART-B: Mini Project (Max. Exam Mks. 30)**

- Use Java, C#, PHP, Python, or any other similar front-end tool. All applications must be demonstrated on desktop/laptop as a stand-alone or web based application (Mobile apps on Android/IOS are not permitted.)

**Lab Experiments:**

**Part A: SQL Programming**

**1** Consider the following schema for a Library Database:
BOOK(Book_id, Title, Publisher_Name, Pub_Year)
BOOK_AUTHORS(Book_id, Author_Name)
PUBLISHER(Name, Address, Phone)
BOOK_COPIES(Book_id, Branch_id, No-of_Copies)
BOOK_LENDING(Book_id, Branch_id, Card_No, Date_Out, Due_Date)
LIBRARY_BRANCH(Branch_id, Branch_Name, Address)
Write SQL queries to

1. Retrieve details of all books in the library – id, title, name of publisher, authors, number of copies in each branch, etc.
2. Get the particulars of borrowers who have borrowed more than 3 books, but from Jan 2017 to Jun 2017.
3. Delete a book in BOOK table. Update the contents of other tables to reflect this data manipulation operation.
4. Partition the BOOK table based on year of publication. Demonstrate its working with a simple query.
5. Create a view of all books and its number of copies that are currently available in the Library.

**2** Consider the following schema for Order Database:
SALESMAN(Salesman_id, Name, City, Commission)
CUSTOMER(Customer_id, Cust_Name, City, Grade, Salesman_id)
ORDERS(Ord_No, Purchase_Amt, Ord_Date, Customer_id, Salesman_id)
Write SQL queries to

1. Count the customers with grades above Bangalore's average.
2. Find the name and numbers of all salesman who had more than one customer.
3. List all the salesman and indicate those who have and don't have customers in their cities (Use UNION operation.)
4. Create a view that finds the salesman who has the customer with the highest order of a day.

| | |
|---|---|
| | 5. Demonstrate the DELETE operation by removing salesman with id 1000. All his orders must also be deleted. |
| 3 | Consider the schema for Movie Database:<br>ACTOR(Act_id, Act_Name, Act_Gender)<br>DIRECTOR(Dir_id, Dir_Name, Dir_Phone)<br>MOVIES(Mov_id, Mov_Title, Mov_Year, Mov_Lang, Dir_id)<br>MOVIE_CAST(Act_id, Mov_id, Role)<br>RATING(Mov_id, Rev_Stars)<br>Write SQL queries to<br>1. List the titles of all movies directed by 'Hitchcock'.<br>2. Find the movie names where one or more actors acted in two or more movies.<br>3. List all actors who acted in a movie before 2000 and also in a movie after 2015 (use JOIN operation).<br>4. Find the title of movies and number of stars for each movie that has at least one rating and find the highest number of stars that movie received. Sort the result by movie title.<br>5. Update rating of all movies directed by 'Steven Spielberg' to 5. |
| 4 | Consider the schema for College Database:<br>STUDENT(USN, SName, Address, Phone, Gender)<br>SEMSEC(SSID, Sem, Sec)<br>CLASS(USN, SSID)<br>SUBJECT(Subcode, Title, Sem, Credits)<br>IAMARKS(USN, Subcode, SSID, Test1, Test2, Test3, FinalIA)<br>Write SQL queries to<br>1. List all the student details studying in fourth semester 'C' section.<br>2. Compute the total number of male and female students in each semester and in each section.<br>3. Create a view of Test1 marks of student USN '1BI17CS101' in all subjects.<br>4. Calculate the FinalIA (average of best two test marks) and update the corresponding table for all students.<br>5. Categorize students based on the following criterion:<br>If FinalIA = 17 to 20 then CAT = 'Outstanding'<br>If FinalIA = 12 to 16 then CAT = 'Average'<br>If FinalIA< 12 then CAT = 'Weak'<br>Give these details only for 8th semester A, B, and C section students. |
| 5 | Consider the schema for Company Database:<br>EMPLOYEE(SSN, Name, Address, Sex, Salary, SuperSSN, DNo)<br>DEPARTMENT(DNo, DName, MgrSSN, MgrStartDate)<br>DLOCATION(DNo,DLoc)<br>PROJECT(PNo, PName, PLocation, DNo)<br>WORKS_ON(SSN, PNo, Hours)<br>Write SQL queries to<br>1. Make a list of all project numbers for projects that involve an employee whose last name is 'Scott', either as a worker or as a manager of the department that controls the project.<br>2. Show the resulting salaries if every employee working on the 'IoT' project is given a 10 percent raise.<br>3. Find the sum of the salaries of all employees of the 'Accounts' department, as well as the maximum salary, the minimum salary, and the average salary in this department |

4. Retrieve the name of each employee who works on all the projects controlledby department number 5 (use NOT EXISTS operator).
5. For each department that has more than five employees, retrieve the department number and the number of its employees who are making more than Rs. 6,00,000.

**Part B: Mini project**

- For any problem selected, write the ER Diagram, apply ER-mapping rules, normalize the relations, and follow the application development process.
- Make sure that the application should have five or more tables, at least one trigger and one stored procedure, using suitable frontend tool.
- Indicative areas include; health care, education, industry, transport, supply chain, etc.

**Course outcomes:** The students should be able to:

- Use Structured Query Language (SQL) for database Creation and manipulation.
- Demonstrate the working of different concepts of DBMS
- Implement and test the project developed for an application.

**Conduction of Practical Examination:**

1. All laboratory experiments from part A are to be included for practical examination.
2. Mini project has to be evaluated for 40 Marks.
3. Report should be prepared in a standard format prescribed for project work.
4. Students are allowed to pick one experiment from the lot.
5. Strictly follow the instructions as printed on the cover page of answer script.
6. Marks distribution:
   a) Part A: Procedure + Conduction + Viva: **09 + 42 +09 =60 Marks**
7. Part B: Demonstration + Report + Viva voce = **20+14+06 = 40** Marks
8. Change of experiment is allowed only once and marks allotted to the procedure part to be made zero.

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**CRYPTOGRAPHY, NETWORK SECURITY AND CYBER LAW**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 - 2018)**
**SEMESTER – VI**

| | | | |
|---|---|---|---|
| Subject Code | 17CS61 | IA Marks | 40 |
| Number of Lecture Hours/Week | 4 | Exam Marks | 60 |
| Total Number of Lecture Hours | 50 | Exam Hours | 03 |

**CREDITS – 04**

| **Module – 1** | **Teaching Hours** |
|---|---|
| Introduction - Cyber Attacks, Defence Strategies and Techniques, Guiding Principles, Mathematical Background for Cryptography - Modulo Arithmetic's, The Greatest Comma Divisor, Useful Algebraic Structures, Chinese Remainder Theorem, Basics of Cryptography - Preliminaries, Elementary Substitution Ciphers, Elementary Transport Ciphers, Other Cipher Properties, Secret Key Cryptography – Product Ciphers, DES Construction. | **10 Hours** |
| **Module – 2** | |
| Public Key Cryptography and RSA – RSA Operations, Why Does RSA Work?, Performance, Applications, Practical Issues, Public Key Cryptography Standard (PKCS), Cryptographic Hash - Introduction, Properties, Construction, Applications and Performance, The Birthday Attack, Discrete Logarithm and its Applications - Introduction, Diffie-Hellman Key Exchange, Other Applications. | **10 Hours** |
| **Module – 3** | |
| Key Management - Introduction, Digital Certificates, Public Key Infrastructure, Identity–based Encryption, Authentication–I - One way Authentication, Mutual Authentication, Dictionary Attacks, Authentication – II – Centalised Authentication, The Needham-Schroeder Protocol, Kerberos, Biometrics, IPSec-Security at the Network Layer – Security at Different layers: Pros and Cons, IPSec in Action, Internet Key Exchange (IKE) Protocol, Security Policy and IPSEC, Virtual Private Networks, Security at the Transport Layer - Introduction, SSL Handshake Protocol, SSL Record Layer Protocol, OpenSSL. | **10 Hours** |
| **Module – 4** | |
| IEEE 802.11 Wireless LAN Security - Background, Authentication, Confidentiality and Integrity, Viruses, Worms, and Other Malware, Firewalls – Basics, Practical Issues, Intrusion Prevention and Detection - Introduction, Prevention Versus Detection, Types of Instruction Detection Systems, DDoS Attacks Prevention/Detection, Web Service Security – Motivation, Technologies for Web Services, WS- Security, SAML, Other Standards. | **10 Hours** |
| **Module – 5** | |
| IT act aim and objectives, Scope of the act, Major Concepts, Important provisions, Attribution, acknowledgement, and dispatch of electronic records, Secure electronic records and secure digital signatures, Regulation of certifying authorities: Appointment of Controller and Other officers, Digital Signature certificates, Duties of Subscribers, Penalties and adjudication, The cyber regulations appellate tribunal, Offences, Network service providers not to be liable in certain cases, Miscellaneous Provisions. | **10 Hours** |

**Course outcomes:** The students should be able to:

- Discuss the cryptography and its need to various applications
- Design and Develop simple cryptography algorithms

| |
|---|
| • Understand the cyber security and need cyber Law |
| **Question paper pattern:**<br>The question paper will have TEN questions.<br>There will be TWO questions from each module.<br>Each question will have questions covering all the topics under a module.<br>The students will have to answer FIVE full questions, selecting ONE full question from each module. |
| **Text Books:** |
| 1. Cryptography, Network Security and Cyber Laws – Bernard Menezes, Cengage Learning, 2010 edition (Chapters-1,3,4,5,6,7,8,9,10,11,12,13,14,15,19(19.1-19.5),21(21.1-21.2),22(22.1-22.4),25 |
| **Reference Books:** |
| 1. Cryptography and Network Security- Behrouz A Forouzan, DebdeepMukhopadhyay, Mc-GrawHill, 3rd Edition, 2015<br>2. Cryptography and Network Security- William Stallings, Pearson Education, 7th Edition<br>3. Cyber Law simplified- VivekSood, Mc-GrawHill, 11th reprint , 2013<br>4. Cyber security and Cyber Laws, Alfred Basta, Nadine Basta, Mary brown, ravindrakumar, Cengage learning |

Dept. Of Co... Engineering
Alva's Institute of ... & Technology
Mijar, MOODBIDRI - 574 225

## COMPUTER GRAPHICS AND VISUALIZATION
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 - 2018)**
**SEMESTER – VI**

| | | | |
|---|---|---|---|
| Subject Code | 17CS62 | IA Marks | 40 |
| Number of Lecture Hours/Week | 4 | Exam Marks | 60 |
| Total Number of Lecture Hours | 50 | Exam Hours | 03 |

**CREDITS – 04**

| **Module – 1** | **Teaching Hours** |
|---|---|
| **Overview: Computer Graphics and OpenGL:** Computer Graphics:Basics of computer graphics, Application of Computer Graphics, Video Display Devices: Random Scan and Raster Scan displays, color CRT monitors, Flat panel displays. Raster-scan systems: video controller, raster scan Display processor, graphics workstations and viewing systems, Input devices, graphics networks, graphics on the internet, graphics software. OpenGL: Introduction to OpenGL ,coordinate reference frames, specifying two-dimensional world coordinate reference frames in OpenGL, OpenGL point functions, OpenGL line functions, point attributes, line attributes, curve attributes, OpenGL point attribute functions, OpenGL line attribute functions, Line drawing algorithms(DDA, Bresenham's), circle generation algorithms(Bresenham's).<br>**Text-1:Chapter -1: 1-1 to 1-9,2-1 to 2-9 (Excluding 2-5),3-1 to 3-5,3-9,3-20** | **10 Hours** |
| **Module – 2** | |
| **Fill area Primitives, 2D Geometric Transformations and 2D viewing:** Fill area Primitives: Polygon fill-areas, OpenGL polygon fill area functions, fill area attributes, general scan line polygon fill algorithm, OpenGL fill-area attribute functions. 2DGeometric Transformations: Basic 2D Geometric Transformations, matrix representations and homogeneous coordinates. Inverse transformations, 2DComposite transformations, other 2D transformations, raster methods for geometric transformations, OpenGL raster transformations, OpenGL geometric transformations function, 2D viewing: 2D viewing pipeline, OpenGL 2D viewing functions.<br>**Text-1:Chapter 3-14 to 3-16,4-9,4-10,4-14,5-1 to 5-7,5-17,6-1,6-4** | **10 Hours** |
| **Module – 3** | |
| **Clipping,3D Geometric Transformations, Color and Illumination Models:** Clipping: clipping window, normalization and viewport transformations, clipping algorithms,2D point clipping, 2D line clipping algorithms: cohen-sutherland line clipping only -polygon fill area clipping: Sutherland-Hodgeman polygon clipping algorithm only.3DGeometric Transformations: 3D translation, rotation, scaling, composite 3D transformations, other 3D transformations, affine transformations, OpenGL geometric transformations functions. Color Models: Properties of light, color models, RGB and CMY color models. Illumination Models: Light sources, basic illumination models-Ambient light, diffuse reflection, specular and phong model, Corresponding openGL functions.<br>**Text-1:Chapter :6-2 to 6-08 (Excluding 6-4),5-9 to 5-17(Excluding 5-15),12-1,12-2,12-4,12-6,10-1,10-3** | **10 Hours** |
| **Module – 4** | |
| **3D Viewing and Visible Surface Detection:** 3DViewing:3D viewing concepts, 3D viewing pipeline, 3D viewing coordinate parameters , Transformation from | **10 Hours** |

| | |
|---|---|
| world to viewing coordinates, Projection transformation, orthogonal projections, perspective projections, The viewport transformation and 3D screen coordinates. OpenGL 3D viewing functions. Visible Surface Detection Methods: Classification of visible surface Detection algorithms, back face detection, depth buffer method and OpenGL visibility detection functions.<br>**Text-1:Chapter: 7-1 to 7-10(Excluding 7-7), 9-1 to 9-3, 9-14** | |
| **Module – 5** | |
| **Input& interaction, Curves and Computer Animation:** Input and Interaction: Input devices, clients and servers, Display Lists, Display Lists and Modelling, Programming Event Driven Input, Menus Picking, Building Interactive Models, Animating Interactive programs, Design of Interactive programs, Logic operations .Curved surfaces, quadric surfaces, OpenGL Quadric-Surface and Cubic-Surface Functions, Bezier Spline Curves, Bezier surfaces, OpenGL curve functions. Corresponding openGL functions.<br>**Text-1:Chapter :8-3 to 8-6 (Excluding 8-5),8-9,8-10,8-11,3-8,8-18,13-11,3-2,13-3,13-4,13-10**<br>**Text-2:Chapter 3: 3-1 to 3.11: Input& interaction** | **10 Hours** |
| **Course outcomes:** The students should be able to: | |
| • Design and implement algorithms for 2D graphics primitives and attributes.<br>• Illustrate Geometric transformations on both 2D and 3D objects.<br>• Understand the concepts of clipping and visible surface detection in 2D and 3D viewing, and Illumination Models.<br>• Discussabout suitable hardware and software for developing graphics packages using OpenGL. | |
| **Question paper pattern:**<br>The question paper will have TEN questions.<br>There will be TWO questions from each module.<br>Each question will have questions covering all the topics under a module.<br>The students will have to answer FIVE full questions, selecting ONE full question from each module. | |
| **Text Books:** | |
| 1. Donald Hearn & Pauline Baker: Computer Graphics with OpenGL Version,3rd/ 4thEdition, Pearson Education,2011<br>2. Edward Angel: Interactive Computer Graphics- A Top Down approach with OpenGL, 5th edition. Pearson Education, 2008 | |
| **Reference Books:** | |
| 1. James D Foley, Andries Van Dam, Steven K Feiner, John F Huges Computer graphics with OpenGL: pearson education<br>2. Xiang, Plastock : Computer Graphics , sham's outline series, 2nd edition, TMG.<br>3. Kelvin Sung, Peter Shirley, steven Baer : Interactive Computer Graphics, concepts and applications, Cengage Learning<br>4. M MRaiker, Computer Graphics using OpenGL, Filip learning/Elsevier | |

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**SYSTEM SOFTWARE AND COMPILER DESIGN**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 - 2018)**
**SEMESTER – VI**

| | | | |
|---|---|---|---|
| Subject Code | 17CS63 | IA Marks | 40 |
| Number of Lecture Hours/Week | 4 | Exam Marks | 60 |
| Total Number of Lecture Hours | 50 | Exam Hours | 03 |

**CREDITS – 04**

| **Module – 1** | **Teaching Hours** |
|---|---|
| Introduction to System Software, Machine Architecture of SIC and SIC/XE. **Assemblers:** Basic assembler functions, machine dependent assembler features, machine independent assembler features, assembler design options. **Macroprocessors:** Basicmacro processor functions, **Text book 1: Chapter 1: 1.1,1.2,1.3.1,1.3.2, Chapter2 : 2.1-2.4,Chapter4: 4.1.1,4.1.2** | **10 Hours** |
| **Module – 2** | |
| **Loaders and Linkers:** Basic Loader Functions, Machine Dependent Loader Features, Machine Independent Loader Features, Loader Design Options, Implementation Examples. **Text book 1 : Chapter 3 ,3.1 -3.5** | **10 Hours** |
| **Module – 3** | |
| **Introduction:** Language Processors, The structure of a compiler, The evaluation of programming languages, The science of building compiler, Applications of compiler technology, Programming language basics **Lexical Analysis:** The role of lexical analyzer, Input buffering, Specifications of token, recognition of tokens, lexical analyzer generator, Finite automate. **Text book 2:Chapter 1 1.1-1.6 Chapter 3 3.1 – 3.6** | **10 Hours** |
| **Module – 4** | |
| Syntax Analysis: Introduction, Role Of Parsers, Context Free Grammars, Writing a grammar, Top Down Parsers, Bottom-Up Parsers, Operator-Precedence Parsing **Text book 2: Chapter 4 4.1 4.2 4.3 4.4 4.5 4.6 Text book 1 : 5.1.3** | **10 Hours** |
| **Module – 5** | |
| Syntax Directed Translation, Intermediate code generation, Code generation **Text book 2: Chapter 5.1, 5.2, 5.3, 6.1, 6.2, 8.1, 8.2** | **10 Hours** |

**Course outcomes:** The students should be able to:

- Illustrate system software such as assemblers, loaders, linkers and macroprocessors
- Design and develop lexical analyzers, parsers and code generators
- Discuss about lex and yacc tools for implementing different concepts of system software

**Question paper pattern:**
The question paper will have TEN questions.
There will be TWO questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer FIVE full questions, selecting ONE full question from each module.

**Text Books:**

1. System Software by Leland. L. Beck, D Manjula, 3rd edition, 2012

| |
|---|
| 2. Compilers-Principles, Techniques and Tools by Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman. Pearson, 2nd edition, 2007 |
| **Reference Books:** |
| 1. Systems programming – Srimanta Pal , Oxford university press, 2016<br>2. System programming and Compiler Design, K C Louden, Cengage Learning<br>3. System software and operating system by D. M. Dhamdhere TMG<br>4. Compiler Design, K Muneeswaran, Oxford University Press 2013. |

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

| OPERATING SYSTEMS<br>[As per Choice Based Credit System (CBCS) scheme]<br>(Effective from the academic year 2017 - 2018)<br>SEMESTER – VI | | | |
|---|---|---|---|
| Subject Code | 17CS64 | IA Marks | 40 |
| Number of Lecture Hours/Week | 4 | Exam Marks | 60 |
| Total Number of Lecture Hours | 50 | Exam Hours | 03 |
| CREDITS – 04 | | | |

| Module – 1 | Teaching Hours |
|---|---|
| **Introduction to operating systems, System structures:** What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating System operations; Process management; Memory management; Storage management; Protection and Security; Distributed system; Special-purpose systems; Computing environments. Operating System Services; User - Operating System interface; System calls; Types of system calls; System programs; Operating system design and implementation; Operating System structure; Virtual machines; Operating System generation; System boot. **Process Management** Process concept; Process scheduling; Operations on processes; Inter process communication | **10 Hours** |
| **Module – 2** | |
| **Multi-threaded Programming**: Overview; Multithreading models; Thread Libraries; Threading issues. Process Scheduling: Basic concepts; Scheduling Criteria; Scheduling Algorithms; Multiple-processor scheduling; Thread scheduling. **Process Synchronization:** Synchronization: The critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors. | **10 Hours** |
| **Module – 3** | |
| **Deadlocks :** Deadlocks; System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock. **Memory Management:** Memory management strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. | **10 Hours** |
| **Module – 4** | |
| **Virtual Memory Management**: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing. **File System, Implementation of File System:** File system: File concept; Access methods; Directory structure; File system mounting; File sharing; Protection: Implementing File system: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management. | **10 Hours** |
| **Module – 5** | |
| **Secondary Storage Structures, Protection:** Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management; Swap space management. Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights, Capability- Based systems. **Case Study: The Linux Operating System:** Linux history; Design principles; Kernel modules; Process management; Scheduling; Memory Management; File systems, Input and output; | **10 Hours** |

| |
|---|
| Inter-process communication. |
| **Course outcomes:** The students should be able to: |
| • Demonstrate need for OS and different types of OS<br>• Discuss suitable techniques for management of different resources<br>• Illustrate processor, memory, storage and file system commands<br>• Explain the different concepts of OS in platform of usage through case studies |
| **Question paper pattern:**<br>The question paper will have TEN questions.<br>There will be TWO questions from each module.<br>Each question will have questions covering all the topics under a module.<br>The students will have to answer FIVE full questions, selecting ONE full question from each module. |
| **Text Books:** |
| 1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Principles 7th edition, Wiley-India, 2006. |
| **Reference Books** |
| 1. Ann McHoes Ida M Fylnn, Understanding Operating System, Cengage Learning, 6th Edition<br>2. D.M Dhamdhere, Operating Systems: A Concept Based Approach 3rd Ed, McGraw-Hill, 2013.<br>3. P.C.P. Bhatt, An Introduction to Operating Systems: Concepts and Practice 4th Edition, PHI(EEE), 2014.<br>4. William Stallings Operating Systems: Internals and Design Principles, 6th Edition, Pearson. |

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## OPERATIONS RESEARCH

**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 - 2018)**
**SEMESTER – VI**

| | | | |
|---|---|---|---|
| Subject Code | 17CS653 | IA Marks | 40 |
| Number of Lecture Hours/Week | 3 | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 03**

| **Module – 1** | **Teaching Hours** |
|---|---|
| **Introduction, Linear Programming:** Introduction: The origin, natureand impact of OR; Defining the problem and gathering data; Formulating amathematical model; Deriving solutions from the model; Testing the model;Preparing to apply the model; Implementation .<br>**Introduction to Linear Programming Problem (LPP):** Prototype example, Assumptions of LPP, Formulation of LPP and Graphical method various examples. | **8 Hours** |
| **Module – 2** | |
| **Simplex Method – 1:** The essence of the simplex method; Setting up the simplex method; Types of variables, Algebraof the simplex method; the simplex method in tabular form; Tie breaking inthe simplex method, Big M method, Two phase method. | **8 Hours** |
| **Module – 3** | |
| **Simplex Method – 2: Duality Theory** - The essence of duality theory, Primaldual relationship, conversion of primal to dual problem and vice versa. The dual simplex method. | **8 Hours** |
| **Module – 4** | |
| **Transportation and Assignment Problems:** The transportation problem, Initial Basic Feasible Solution (IBFS) by North West Corner Rule method, Matrix Minima Method, Vogel's Approximation Method. Optimal solution by Modified Distribution Method (MODI). The Assignment problem; A Hungarian algorithm for the assignment problem. Minimization and Maximization varieties in transportation and assignment problems. | **8 Hours** |
| **Module – 5** | |
| **Game Theory:** Game Theory: The formulation of twopersons, zero sum games; saddle point, maximin and minimax principle, Solving simple games- a prototype example;Games with mixed strategies; Graphical solution procedure.<br>**Metaheuristics:** The nature of Metaheuristics, Tabu Search, SimulatedAnnealing, Genetic Algorithms. | **8 Hours** |

**Course outcomes:** The students should be able to:

- Explain optimization techniques for various problems.
- Understand the given problem as transportation and assignment problem and solve.
- Illustrate game theory for decision support system.

**Question paper pattern:**
The question paper will have TEN questions.
There will be TWO questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer FIVE full questions, selecting ONE full question from each module.

| Text Books: |
| --- |
| 1. D.S. Hira and P.K. Gupta, Operations Research, (Revised Edition), Published by S. Chand & Company Ltd, 2014 |
| **Reference Books:** |
| 1. S Kalavathy, Operation Research, Vikas Publishing House Pvt Limited, 01-Aug-2002<br>2. S D Sharma, Operation Research,KedarNath Ram Nath Publishers. |

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## MOBILE APPLICATION DEVELOPMENT
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 -2018)**
**SEMESTER – VI**

| | | | |
|---|---|---|---|
| Subject Code | 17CS661 | IA Marks | 40 |
| Number of Lecture Hours/Week | 3 | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 03**

| **Module – 1** | **Teaching Hours** |
|---|---|
| Get started, Build your first app, Activities, Testing, debugging and using support libraries | **8 Hours** |
| **Module – 2** | |
| User Interaction, Delightful user experience, Testing your UI | **8 Hours** |
| **Module – 3** | |
| Background Tasks, Triggering, scheduling and optimizing background tasks | **8 Hours** |
| **Module – 4** | |
| All about data, Preferences and Settings, Storing data using SQLite, Sharing data with content providers, Loading data using Loaders | **8 Hours** |
| **Module – 5** | |
| Permissions, Performance and Security, Firebase and AdMob, Publish | **8 Hours** |

**Course outcomes:** The students should be able to:

- Design and Develop Android application by setting up Android development environment
- Implement adaptive, responsive user interfaces that work across a wide range of devices.
- Explainlong running tasks and background work in Android applications
- Demonstrate methods in storing, sharing and retrieving data in Android applications
- Discuss the performance of android applications and understand the role of permissions and security
- Describe the steps involved in publishing Android application to share with the world

**Question paper pattern:**
The question paper will have TEN questions.
There will be TWO questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer FIVE full questions, selecting ONE full question from each module.

**Text Books:**


1. Google Developer Training, "Android Developer Fundamentals Course – Concept Reference", Google Developer Training Team, 2017. https://www.gitbook.com/book/google-developer-training/android-developer-fundamentals-course-concepts/details (Download pdf file from the above link)


**Reference Books:**


1. Erik Hellman, "Android Programming – Pushing the Limits", 1st Edition, Wiley India Pvt Ltd, 2014.
2. Dawn Griffiths and David Griffiths, "Head First Android Development", 1st Edition, O'Reilly SPD Publishers, 2015.
3. J F DiMarzio, "Beginning Android Programming with Android Studio", 4th Edition,

Wiley India Pvt Ltd, 2016. ISBN-13: 978-8126565580
4. AnubhavPradhan, Anil V Deshpande, " Composing Mobile Apps" using Android, Wiley 2014, ISBN: 978-81-265-4660-2

H.O.D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## PYTHON APPLICATION PROGRAMMING

**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 -2018)**
**SEMESTER – VI**

| | | | |
|---|---|---|---|
| Subject Code | 17CS664 | IA Marks | 40 |
| Number of Lecture Hours/Week | 3 | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 03**

| **Module – 1** | **Teaching Hours** |
|---|---|
| Why should you learn to write programs, Variables, expressions and statements, Conditional execution, Functions | **8 Hours** |
| **Module – 2** | |
| Iteration, Strings, Files | **8 Hours** |
| **Module – 3** | |
| Lists, Dictionaries, Tuples, Regular Expressions | **8 Hours** |
| **Module – 4** | |
| Classes and objects, Classes and functions, Classes and methods | **8 Hours** |
| **Module – 5** | |
| Networked programs, Using Web Services, Using databases and SQL | **8 Hours** |

**Course outcomes:** The students should be able to:

- Understand Python syntax and semantics and be fluent in the use of Python flow control and functions.
- Demonstrate proficiency in handling Strings and File Systems.
- Implement Python Programs using core data structures like Lists, Dictionaries and use Regular Expressions.
- Interpret the concepts of Object-Oriented Programming as used in Python.
- Implement exemplary applications related to Network Programming, Web Services and Databases in Python.

**Question paper pattern:**
The question paper will have TEN questions.
There will be TWO questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer FIVE full questions, selecting ONE full question from each module.

**Text Books:**


1. Charles R. Severance, "Python for Everybody: Exploring Data Using Python 3", 1st Edition, CreateSpace Independent Publishing Platform, 2016. (http://do1.dr-chuck.com/pythonlearn/EN_us/pythonlearn.pdf ) (Chapters 1 – 13, 15)
2. Allen B. Downey, "Think Python: How to Think Like a Computer Scientist", 2nd Edition, Green Tea Press, 2015. (http://greenteapress.com/thinkpython2/thinkpython2.pdf) (Chapters 15, 16, 17)(Download pdf files from the above links)


**Reference Books:**


1. Charles Dierbach, "Introduction to Computer Science Using Python", 1st Edition, Wiley India Pvt Ltd. ISBN-13: 978-8126556014
2. Mark Lutz, "Programming Python", 4th Edition, O'Reilly Media, 2011.ISBN-13: 978-9350232873

3. Wesley J Chun, "Core Python Applications Programming", 3rd Edition,Pearson Education India, 2015. ISBN-13: 978-9332555365
4. Roberto Tamassia, Michael H Goldwasser, Michael T Goodrich, "Data Structures and Algorithms in Python",1st Edition, Wiley India Pvt Ltd, 2016. ISBN-13: 978-8126562176
5. ReemaThareja, "Python Programming using problem solving approach", Oxford university press, 2017

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**SYSTEM SOFTWARE AND OPERATING SYSTEM LABORATORY**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 - 2018)**
**SEMESTER – VI**

| | | | |
|---|---|---|---|
| Subject Code | 17CSL67 | IA Marks | 40 |
| Number of Lecture Hours/Week | 01I + 02P | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 02**

**Description (If any):**

Exercises to be prepared with minimum three files (Where ever necessary):

i. Header file.
ii. Implementation file.
iii. Application file where main function will be present.

The idea behind using three files is to differentiate between the developer and user sides. In the developer side, all the three files could be made visible. For the user side only header file and application files could be made visible, which means that the object code of the implementation file could be given to the user along with the interface given in the header file, hiding the source file, if required. Avoid I/O operations (printf/scanf) and use ***data input file*** where ever it is possible

**Lab Experiments:**

1.

a) Write a LEX program to recognize valid ***arithmetic expression.*** Identifiers in the expression could be only integers and operators could be + and *. Count the identifiers & operators present and print them separately.

b) Write YACC program to evaluate ***arithmetic expression*** involving operators: +, -, *, and /

2. Develop, Implement and Execute a program using YACC tool to recognize all strings ending with ***b*** preceded by ***na's*** using the grammar $a^n b$ (note: input ***n*** value)

3. Design, develop and implement YACC/C program to construct ***Predictive / LL(1) Parsing Table*** for the grammar rules: $A \rightarrow aBa$ , $B \rightarrow bB \mid \varepsilon$. Use this table to parse the sentence: ***abba$***

4. Design, develop and implement YACC/C program to demonstrate ***Shift Reduce Parsing*** techniquefor the grammar rules: $E \rightarrow E+T \mid T$, $T \rightarrow T*F \mid F$, $F \rightarrow (E) \mid id$ and parse the sentence: ***id + id * id***.

5. Design, develop and implement a C/Java program to generate the machine code using ***Triples*** for the statement $A = -B * (C + D)$ whose intermediate code in three-address form:

$T1 = -B$
$T2 = C + D$
$T3 = T1 + T2$
$A = T3$

6. a) Write a LEX program to eliminate ***comment lines*** in a ***C*** program and copy the

resulting program into a separate file.

b) Write YACC program to recognize valid ***identifier, operators and keywords*** in the given text (*C program*) file.

7. Design, develop and implement a C/C++/Java program to simulate the working of Shortest remaining time and Round Robin (RR) scheduling algorithms. Experiment with different quantum sizes for RR algorithm.
8. Design, develop and implement a C/C++/Java program to implement Banker's algorithm. Assume suitable input required to demonstrate the results.
9. Design, develop and implement a C/C++/Java program to implement page replacement algorithms LRU and FIFO. Assume suitable input required to demonstrate the results.

**Study Experiment / Project:**

**NIL**

**Course outcomes:** The students should be able to:

- Implement and demonstrate Lexer's and Parser's
- Implement different algorithms required for management, scheduling, allocation and communication used in operating system.

**Conduction of Practical Examination:**

- All laboratory experiments are to be included for practical examination.
- Students are allowed to pick one experiment from the lot.
- Strictly follow the instructions as printed on the cover page of answer script
- Marks distribution: Procedure + Conduction + Viva:**15 + 70 +15 (100)**
- **Change of experiment is allowed only once and marks allotted to the procedure part to be made zero**

H.O.D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## COMPUTER GRAPHICS LABORATORY WITH MINI PROJECT

**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 - 2018)**
**SEMESTER – VI**

| | | | |
|---|---|---|---|
| Subject Code | 17CSL68 | IA Marks | 40 |
| Number of Lecture Hours/Week | 01I + 02P | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 02**

**Description (If any):**

--

**Lab Experiments:**

**PART A**

**Design, develop, and implement the following programs using OpenGL API**

1. Implement Brenham's line drawing algorithm for all types of slope.
   **Refer:Text-1: Chapter 3.5**
   **Refer:Text-2: Chapter 8**
2. Create and rotate a triangle about the origin and a fixed point.
   **Refer:Text-1: Chapter 5-4**
3. Draw a colour cube and spin it using OpenGL transformation matrices.
   **Refer:Text-2: Modelling a Coloured Cube**
4. Draw a color cube and allow the user to move the camera suitably to experiment with perspective viewing.
   **Refer:Text-2: Topic: Positioning of Camera**
5. Clip a lines using Cohen-Sutherland algorithm
   **Refer:Text-1: Chapter 6.7**
   **Refer:Text-2: Chapter 8**
6. To draw a simple shaded scene consisting of a tea pot on a table. Define suitably the position and properties of the light source along with the properties of the surfaces of the solid object used in the scene.
   **Refer:Text-2: Topic: Lighting and Shading**
7. Design, develop and implement recursively subdivide a tetrahedron to form 3D sierpinski gasket. The number of recursive steps is to be specified by the user.
   **Refer: Text-2: Topic:**sierpinski gasket.
8. Develop a menu driven program to animate a flag using Bezier Curve algorithm
   **Refer: Text-1: Chapter 8-10**
9. Develop a menu driven program to fill the polygon using scan line algorithm

**Project:**

**PART –B ( MINI-PROJECT) :**

Student should develop mini project on the topics mentioned below or similar applications using Open GL API. Consider all types of attributes like color, thickness, styles, font, background, speed etc., while doing mini project.

**(During the practical exam: the students should demonstrate and answer Viva-Voce)**

**Sample Topics:**

**Simulation of concepts of OS, Data structures, algorithms etc.**

**Course outcomes:** The students should be able to:

- Apply the concepts of computer graphics
- Implement computer graphics applications using OpenGL
- Implement real world problems using OpenGL

**Conduction of Practical Examination:**

1. All laboratory experiments from part A are to be included for practical examination.
2. Mini project has to be evaluated for 40 Marks.
3. Report should be prepared in a standard format prescribed for project work.
4. Students are allowed to pick one experiment from the lot.
5. Strictly follow the instructions as printed on the cover page of answer script.
6. Marks distribution:
   a) Part A: Procedure + Conduction + Viva: **09 + 42 +09 =60 Marks**
   b) Part B: Demonstration + Report + Viva voce = **20+14+06 = 40 Marks**
7. Change of experiment is allowed only once and marks allotted to the procedure part to be made zero.

**Reference books:**


1. Donald Hearn & Pauline Baker: Computer Graphics-OpenGL Version,3rd Edition, Pearson Education,2011
2. Edward Angel: Interactive computer graphics- A Top Down approach with OpenGL, 5th edition. Pearson Education, 2011
3. M MRaikar, Computer Graphics using OpenGL, Fillip Learning / Elsevier, Bangalore / New Delhi (2013)


H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## WEB TECHNOLOGY AND ITS APPLICATIONS
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 - 2018)**
**SEMESTER – VII**

| | | | |
|---|---|---|---|
| Subject Code | 17CS71 | IA Marks | 40 |
| Number of Lecture Hours/Week | 04 | Exam Marks | 60 |
| Total Number of Lecture Hours | 50 | Exam Hours | 03 |

**CREDITS – 04**

| **Module – 1** | **Teaching Hours** |
|---|---|
| Introduction to HTML, What is HTML and Where did it come from?, HTML Syntax, Semantic Markup, Structure of HTML Documents, Quick Tour of HTML Elements, HTML5 Semantic Structure Elements, Introduction to CSS, What is CSS, CSS Syntax, Location of Styles, Selectors, The Cascade: How Styles Interact, The Box Model, CSS Text Styling. | **10 Hours** |
| **Module – 2** | |
| HTML Tables and Forms, Introducing Tables, Styling Tables, Introducing Forms, Form Control Elements, Table and Form Accessibility, Microformats, Advanced CSS: Layout, Normal Flow, Positioning Elements, Floating Elements, Constructing Multicolumn Layouts, Approaches to CSS Layout, Responsive Design, CSS Frameworks. | **10 Hours** |
| **Module – 3** | |
| JavaScript: Client-Side Scripting, What is JavaScript and What can it do?, JavaScript Design Principles, Where does JavaScript Go?, Syntax, JavaScript Objects, The Document Object Model (DOM), JavaScript Events, Forms, Introduction to Server-Side Development with PHP, What is Server-Side Development, A Web Server's Responsibilities, Quick Tour of PHP, Program Control, Functions | **10 Hours** |
| **Module – 4** | |
| PHP Arrays and Superglobals, Arrays, $_GET and $_POST Superglobal Arrays, $_SERVER Array, $_Files Array, Reading/Writing Files, PHP Classes and Objects, Object-Oriented Overview, Classes and Objects in PHP, Object Oriented Design, Error Handling and Validation, What are Errors and Exceptions?, PHP Error Reporting, PHP Error and Exception Handling | **10 Hours** |
| **Module – 5** | |
| Managing State, The Problem of State in Web Applications, Passing Information via Query Strings, Passing Information via the URL Path, Cookies, Serialization, Session State, HTML5 Web Storage, Caching, Advanced JavaScript and jQuery, JavaScript Pseudo-Classes, jQuery Foundations, AJAX, Asynchronous File Transmission, Animation, Backbone MVC Frameworks, XML Processing and Web Services, XML Processing, JSON, Overview of Web Services. | **10 Hours** |

**Course Outcomes:** After studying this course, students will be able to

- Define HTML and CSS syntax and semantics to build web pages.
- Understand the concepts of Construct , visually format tables and forms using HTML using CSS
- Develop Client-Side Scripts using JavaScript and Server-Side Scripts using PHP to generate and display the contents dynamically.
- List the principles of object oriented development using PHP
- Illustrate JavaScript frameworks like jQuery and Backbone which facilitates

| |
|---|
| developer to focus on core features. |
| **Question paper pattern:** |
| The question paper will have ten questions.<br>There will be 2 questions from each module.<br>Each question will have questions covering all the topics under a module.<br>The students will have to answer 5 full questions, selecting one full question from each module. |
| **Text Books:** |
| 1. Randy Connolly, Ricardo Hoar, **"Fundamentals of Web Development"**, 1st Edition, Pearson Education India. (**ISBN:**978-9332575271) |
| **Reference Books:** |
| 1) Robin Nixon, **"Learning PHP, MySQL &JavaScript with jQuery, CSS and HTML5"**, 4th Edition, O'Reilly Publications, 2015. (**ISBN:**978-9352130153)<br>2) Luke Welling, Laura Thomson, **"PHP and MySQL Web Development"**, 5th Edition, Pearson Education, 2016. (**ISBN:**978-9332582736)<br>3) Nicholas C Zakas, **"Professional JavaScript for Web Developers"**, 3rd Edition, Wrox/Wiley India, 2012. (**ISBN:**978-8126535088)<br>4) David Sawyer Mcfarland, **"JavaScript & jQuery: The Missing Manual"**, 1st Edition, O'Reilly/Shroff Publishers & Distributors Pvt Ltd, 2014 (**ISBN:**978-9351108078)<br>5) Zak Ruvalcaba Anne Boehm, **"Murach's HTML5 and CSS3"**, 3rdEdition, Murachs/Shroff Publishers & Distributors Pvt Ltd, 2016. (**ISBN:**978-9352133246) |

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## ADVANCED COMPUTER ARCHITECTURES
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 - 2018)**
**SEMESTER – VII**

| | | | |
|---|---|---|---|
| Subject Code | 17CS72 | IA Marks | 40 |
| Number of Lecture Hours/Week | 4 | Exam Marks | 60 |
| Total Number of Lecture Hours | 50 | Exam Hours | 03 |

**CREDITS – 04**

| **Module – 1** | **Teaching Hours** |
|---|---|
| Theory of Parallelism: Parallel Computer Models, The State of Computing, Multiprocessors and Multicomputer ,Multivector and SIMD Computers ,PRAM and VLSI Models, Program and Network Properties ,Conditions of Parallelism, Program Partitioning and Scheduling, Program Flow Mechanisms, System Interconnect Architectures, Principles of Scalable Performance, Performance Metrics and Measures, Parallel Processing Applications, Speedup Performance Laws, Scalability Analysis and Approaches. | **10 Hours** |
| **Module – 2** | |
| Hardware Technologies: Processors and Memory Hierarchy, Advanced Processor Technology, Superscalar and Vector Processors, Memory Hierarchy Technology, Virtual Memory Technology. | **10 Hours** |
| **Module – 3** | |
| Bus, Cache, and Shared Memory ,Bus Systems ,Cache Memory Organizations ,Shared Memory Organizations ,Sequential and Weak Consistency Models ,Pipelining and Superscalar Techniques ,Linear Pipeline Processors ,Nonlinear Pipeline Processors ,Instruction Pipeline Design ,Arithmetic Pipeline Design (Upto 6.4). | **10 Hours** |
| **Module – 4** | |
| Parallel and Scalable Architectures: Multiprocessors and Multicomputers ,Multiprocessor System Interconnects, Cache Coherence and Synchronization Mechanisms, Three Generations of Multicomputers ,Message-Passing Mechanisms ,Multivector and SIMD Computers ,Vector Processing Principles ,Multivector Multiprocessors ,Compound Vector Processing ,SIMD Computer Organizations (Upto 8.4),Scalable, Multithreaded, and Dataflow Architectures, Latency-Hiding Techniques, Principles of Multithreading, Fine-Grain Multicomputers, Scalable and Multithreaded Architectures, Dataflow and Hybrid Architectures. | **10 Hours** |
| **Module – 5** | |
| Software for parallel programming: Parallel Models, Languages, and Compilers ,Parallel Programming Models, Parallel Languages and Compilers ,Dependence Analysis of Data Arrays ,Parallel Program Development and Environments, Synchronization and Multiprocessing Modes. Instruction and System Level Parallelism, Instruction Level Parallelism ,Computer Architecture ,Contents, Basic Design Issues ,Problem Definition ,Model of a Typical Processor ,Compiler-detected Instruction Level Parallelism ,Operand Forwarding ,Reorder Buffer, Register Renaming ,Tomasulo's Algorithm ,Branch Prediction, Limitations in Exploiting Instruction Level Parallelism ,Thread Level Parallelism. | **10 Hours** |
| **Course outcomes:** The students should be able to: | |

| |
|---|
| • Understand the concepts of parallel computing and hardware technologies<br>• Illustrate and contrast the parallel architectures<br>• Recall parallel programming concepts |
| **Question paper pattern**<br>The question paper will have ten questions.<br>There will be 2 questions from each module.<br>Each question will have questions covering all the topics under a module.<br>The students will have to answer 5 full questions, selecting one full question from each module. |
| **Text Books:**<br>1. Kai Hwang and Naresh Jotwani, Advanced Computer Architecture (SIE): Parallelism, Scalability, Programmability, McGraw Hill Education 3/e. 2015 |
| **Reference Books:** |
| 1. John L. Hennessy and David A. Patterson, Computer Architecture: A quantitative approach, 5th edition, Morgan Kaufmann Elseveir, 2013 |

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## MACHINE LEARNING

**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 - 2018)**
**SEMESTER – VII**

| | | | |
|---|---|---|---|
| Subject Code | 17CS73 | IA Marks | 40 |
| Number of Lecture Hours/Week | 03 | Exam Marks | 60 |
| Total Number of Lecture Hours | 50 | Exam Hours | 03 |

**CREDITS – 04**

| **Module – 1** | **Teaching Hours** |
|---|---|
| **Introduction:** Well posed learning problems, Designing a Learning system, Perspective and Issues in Machine Learning.<br>**Concept Learning:** Concept learning task, Concept learning as search, Find-S algorithm, Version space, Candidate Elimination algorithm, Inductive Bias.<br>**Text Book1, Sections: 1.1 – 1.3, 2.1-2.5, 2.7** | **10 Hours** |
| **Module – 2** | |
| **Decision Tree Learning:** Decision tree representation, Appropriate problems for decision tree learning, Basic decision tree learning algorithm, hypothesis space search in decision tree learning, Inductive bias in decision tree learning, Issues in decision tree learning.<br>**Text Book1, Sections: 3.1-3.7** | **10 Hours** |
| **Module – 3** | |
| **Artificial Neural Networks:** Introduction, Neural Network representation, Appropriate problems, Perceptrons, Backpropagation algorithm.<br>**Text book 1, Sections: 4.1 – 4.6** | **08 Hours** |
| **Module – 4** | |
| **Bayesian Learning:** Introduction, Bayes theorem, Bayes theorem and concept learning, ML and LS error hypothesis, ML for predicting probabilities, MDL principle, Naive Bayes classifier, Bayesian belief networks, EM algorithm<br>**Text book 1, Sections: 6.1 – 6.6, 6.9, 6.11, 6.12** | **10 Hours** |
| **Module – 5** | |
| **Evaluating Hypothesis:** Motivation, Estimating hypothesis accuracy, Basics of sampling theorem, General approach for deriving confidence intervals, Difference in error of two hypothesis, Comparing learning algorithms.<br>**Instance Based Learning:** Introduction, k-nearest neighbor learning, locally weighted regression, radial basis function, cased-based reasoning,<br>**Reinforcement Learning:** Introduction, Learning Task, Q Learning<br>**Text book 1, Sections: 5.1-5.6, 8.1-8.5, 13.1-13.3** | **12 Hours** |
| **Course Outcomes:** After studying this course, students will be able to<br>• Recall the problems for machine learning. And select the either supervised, unsupersvised or reinforcement learning.<br>• Understand theory of probability and statistics related to machine learning<br>• Illustrate concept learning, ANN, Bayes classifier, k nearest neighbor, Q, | |
| **Question paper pattern:**<br>The question paper will have ten questions.<br>There will be 2 questions from each module. | |

| |
|---|
| Each question will have questions covering all the topics under a module.<br>The students will have to answer 5 full questions, selecting one full question from each module. |
| **Text Books:** |
| 1. Tom M. Mitchell, Machine Learning, India Edition 2013, McGraw Hill Education. |
| **Reference Books:** |
| 1. Trevor Hastie, Robert Tibshirani, Jerome Friedman, h The Elements of Statistical Learning, 2nd edition, springer series in statistics.<br>2. Ethem Alpaydın, Introduction to machine learning, second edition, MIT press. |

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## UNIX SYSTEM PROGRAMMING

**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 - 2018)**
**SEMESTER – VII**

| | | | |
|---|---|---|---|
| Subject Code | 17CS744 | IA Marks | 40 |
| Number of Lecture Hours/Week | 3 | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 03**

| **Module – 1** | **Teaching Hours** |
|---|---|
| Introduction: UNIX and ANSI Standards: The ANSI C Standard, The ANSI/ISO C++ Standards, Difference between ANSI C and C++, The POSIX Standards, The POSIX.1 FIPS Standard, The X/Open Standards. UNIX and POSIX APIs: The POSIX APIs, The UNIX and POSIX Development Environment, API Common Characteristics. | **8 Hours** |
| **Module – 2** | |
| UNIX Files and APIs: File Types, The UNIX and POSIX File System, The UNIX and POSIX File Attributes, Inodes in UNIX System V, Application Program Interface to Files, UNIX Kernel Support for Files, Relationship of C Stream Pointers and File Descriptors, Directory Files, Hard and Symbolic Links. UNIX File APIs: General File APIs, File and Record Locking, Directory File APIs, Device File APIs, FIFO File APIs, Symbolic Link File APIs. | **8 Hours** |
| **Module – 3** | |
| UNIX Processes and Process Control: The Environment of a UNIX Process: Introduction, main function, Process Termination, Command-Line Arguments, Environment List, Memory Layout of a C Program, Shared Libraries, Memory Allocation, Environment Variables, setjmp and longjmp Functions, getrlimit, setrlimit Functions, UNIX Kernel Support for Processes. Process Control: Introduction, Process Identifiers, fork, vfork, exit, wait, waitpid, wait3, wait4 Functions, Race Conditions, exec Functions, Changing User IDs and Group IDs, Interpreter Files, system Function, Process Accounting, User Identification, Process Times, I/O Redirection. Process Relationships: Introduction, Terminal Logins, Network Logins, Process Groups, Sessions, Controlling Terminal, tcgetpgrp and tcsetpgrp Functions, Job Control, Shell Execution of Programs, Orphaned Process Groups. | **8 Hours** |
| **Module – 4** | |
| Signals and Daemon Processes: Signals: The UNIX Kernel Support for Signals, signal, Signal Mask, sigaction, The SIGCHLD Signal and the waitpid Function, The sigsetjmp and siglongjmp Functions, Kill, Alarm, Interval Timers, POSIX.lb Timers. Daemon Processes: Introduction, Daemon Characteristics, Coding Rules, Error Logging, Client-Server Model. | **8 Hours** |
| **Module – 5** | |
| Interprocess Communication : Overview of IPC Methods, Pipes, popen, pclose Functions, Coprocesses, FIFOs, System V IPC, Message Queues, Semaphores. Shared Memory, Client-Server Properties, Stream Pipes, Passing File Descriptors, An Open Server-Version 1, Client-Server Connection Functions. | **8 Hours** |

**Course outcomes:** The students should be able to:

- Understand the working of Unix Systems
- Illustrate the application/service over a UNIX system.

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

**Text Books:**

1. Unix System Programming Using C++ - Terrence Chan, PHI, 1999.
2. Advanced Programming in the UNIX Environment - W.Richard Stevens, Stephen A. Rago, 3nd Edition, Pearson Education / PHI, 2005.

**Reference Books:**

1. Advanced Unix Programming- Marc J. Rochkind, 2nd Edition, Pearson Education, 2005.
2. The Design of the UNIX Operating System - Maurice.J.Bach, Pearson Education / PHI, 1987.
3. Unix Internals - Uresh Vahalia, Pearson Education, 2001.

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**STORAGE AREA NETWORKS**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 - 2018)**
**SEMESTER – VII**

| | | | |
|---|---|---|---|
| Subject Code | 17CS754 | IA Marks | 40 |
| Number of Lecture Hours/Week | 3 | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 03**

| **Module – 1** | **Teaching Hours** |
|---|---|
| **Storage System** Introduction to evolution of storage architecture, key data center elements, virtualization, and cloud computing. Key data center elements – Host (or compute), connectivity, storage, and application in both classic and virtual environments. RAID implementations, techniques, and levels along with the impact of RAID on application performance.Components of intelligent storage systems and virtual storage provisioning and intelligent storage system implementations. | **8 Hours** |
| **Module – 2** | |
| **Storage Networking Technologies and Virtualization** Fibre Channel SAN components, connectivity options, and topologies including access protection mechanism 'zoning", FC protocol stack, addressing and operations, SAN-based virtualization and VSAN technology, iSCSI and FCIP protocols for storage access over IP network, Converged protocol FCoE and its components, Network Attached Storage (NAS) - components, protocol and operations, File level storage virtualization, Object based storage and unified storage platform. | **8 Hours** |
| **Module – 3** | |
| **Backup, Archive, and Replication** This unit focuses on information availability and business continuity solutions in both virtualized and non-virtualized environments. Business continuity terminologies, planning and solutions, Clustering and multipathing architecture to avoid single points of failure, Backup and recovery - methods, targets and topologies, Data deduplication and backup in virtualized environment, Fixed content and data archive, Local replication in classic and virtual environments, Remote replication in classic and virtual environments, Three-site remote replication and continuous data protection | **8 Hours** |
| **Module – 4** | |
| **Cloud Computing Characteristics and benefits** This unit focuses on the business drivers, definition, essential characteristics, and phases of journey to the Cloud. ,Business drivers for Cloud computing, Definition of Cloud computing, Characteristics of Cloud computing, Steps involved in transitioning from Classic data center to Cloud computing environment Services and deployment models, Cloud infrastructure components, Cloud migration considerations | **8 Hours** |
| **Module – 5** | |
| **Securing and Managing Storage Infrastructure** This chapter focuses on framework and domains of storage security along with covering security. implementation at storage networking. Security threats, and countermeasures in various domains Security solutions for FC-SAN, IP-SAN and NAS environments, Security in virtualized and cloud environments, Monitoring and managing various information infrastructure components in classic and virtual environments, Information lifecycle management (ILM) and storage tiering, | **8 Hours** |

| |
|---|
| Cloud service management activities |
| **Course outcomes:** The students should be able to: |
| • Identify key challenges in managing information and analyze different storage networking technologies and virtualization<br>• Explain components and the implementation of NAS<br>• Describe CAS architecture and types of archives and forms of virtualization<br>• Illustrate the storage infrastructure and management activities |
| **Question paper pattern:**<br>The question paper will have ten questions.<br>There will be 2 questions from each module.<br>Each question will have questions covering all the topics under a module.<br>The students will have to answer 5 full questions, selecting one full question from each module. |
| **Text Books:** |
| 1. Information Storage and Management,Author :EMC Education Services, Publisher: Wiley ISBN: 9781118094839<br>2. Storage Virtualization, Author: Clark Tom, Publisher: Addison Wesley Publishing Company ISBN : 9780321262516 |
| **Reference Books:** |
| NIL |

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## MACHINE LEARNING LABORATORY
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 - 2018)**
**SEMESTER – VII**

| | | | |
|---|---|---|---|
| Subject Code | 17CSL76 | IA Marks | 40 |
| Number of Lecture Hours/Week | 01I + 02P | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 02**

**Description (If any):**

1. The programs can be implemented in either JAVA or Python.
2. For Problems 1 to 6 and 10, programs are to be developed without using the built-in classes or APIs of Java/Python.
3. Data sets can be taken from standard repositories (https://archive.ics.uci.edu/ml/datasets.html) or constructed by the students.

**Lab Experiments:**

1. Implement and demonstratethe **FIND-Salgorithm** for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.
2. For a given set of training data examples stored in a .CSV file, implement and demonstrate the **Candidate-Elimination algorithm**to output a description of the set of all hypotheses consistent with the training examples.
3. Write a program to demonstrate the working of the decision tree based **ID3 algorithm**. Use an appropriate data set for building the decision tree and apply this knowledge toclassify a new sample.
4. Build an Artificial Neural Network by implementing the **Backpropagation algorithm** and test the same using appropriate data sets.
5. Write a program to implement the **naïve Bayesian classifier** for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.
6. Assuming a set of documents that need to be classified, use the **naïve Bayesian Classifier** model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.
7. Write a program to construct a**Bayesian network** considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.
8. Apply **EM algorithm** to cluster a set of data stored in a .CSV file. Use the same data set for clustering using ***k*-Means algorithm**. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.
9. Write a program to implement ***k*-Nearest Neighbour algorithm** to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.
10. Implement the non-parametric **Locally Weighted Regressionalgorithm** in order to fit data points. Select appropriate data set for your experiment and draw graphs.

**Study Experiment / Project:**

**NIL**

**Course outcomes:** The students should be able to:

1. Understand the implementation procedures for the machine learning algorithms.

2. Design Java/Python programs for various Learning algorithms.
3. Apply appropriate data sets to the Machine Learning algorithms.
4. Identify and apply Machine Learning algorithms to solve real world problems.

**Conduction of Practical Examination:**

- All laboratory experiments are to be included for practical examination.
- Students are allowed to pick one experiment from the lot.
- Strictly follow the instructions as printed on the cover page of answer script
- Marks distribution: Procedure + Conduction + Viva:**15 + 70 +15 (100)**

**Change of experiment is allowed only once and marks allotted to the procedure part to be made zero.**

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**WEB TECHNOLOGY LABORATORY WITH MINI PROJECT**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 - 2018)**
**SEMESTER – VII**

| | | | |
|---|---|---|---|
| Subject Code | 17CSL77 | IA Marks | 40 |
| Number of Lecture Hours/Week | 01I + 02P | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 02**

**Description (If any):**

**NIL**

**Lab Experiments:**

**PART A**

1. Write a JavaScript to design a simple calculator to perform the following operations: sum, product, difference and quotient.
2. Write a JavaScript that calculates the squares and cubes of the numbers from 0 to 10 and outputs HTML text that displays the resulting values in an HTML table format.
3. Write a JavaScript code that displays text "TEXT-GROWING" with increasing font size in the interval of 100ms in RED COLOR, when the font size reaches 50pt it displays "TEXT-SHRINKING" in BLUE color. Then the font size decreases to 5pt.
4. Develop and demonstrate a HTML5 file that includes JavaScript script that uses functions for the following problems:
   a. Parameter: A string
   b. Output: The position in the string of the left-most vowel
   c. Parameter: A number
   d. Output: The number with its digits in the reverse order
5. Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, and Name of the College, Branch, Year of Joining, and email id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.
6. Write a PHP program to keep track of the number of visitors visiting the web page and to display this count of visitors, with proper headings.
7. Write a PHP program to display a digital clock which displays the current time of the server.
8. Write the PHP programs to do the following:
   a. Implement simple calculator operations.
   b. Find the transpose of a matrix.
   c. Multiplication of two matrices.
   d. Addition of two matrices.
9. Write a PHP program named states.py that declares a variable states with value "Mississippi Alabama Texas Massachusetts Kansas". write a PHP program that does the following:
   a. Search for a word in variable states that ends in xas. Store this word in element 0 of a list named statesList.

b. Search for a word in states that begins with k and ends in s. Perform a case-insensitive comparison. [Note: Passing re.Ias a second parameter to method compile performs a case-insensitive comparison.] Store this word in element1 of statesList.

c. Search for a word in states that begins with M and ends in s. Store this word in element 2 of the list.

d. Search for a word in states that ends in a. Store this word in element 3 of the list.

10. Write a PHP program to sort the student records which are stored in the database using selection sort.

**Study Experiment / Project:**

Develop a web application project using the languages and concepts learnt in the theory and exercises listed in part A with a good look and feel effects. You can use any web technologies and frameworks and databases.

Note:

1. In the examination each student picks one question from part A.
2. A team of two or three students must develop the mini project. However during the examination, each student must demonstrate the project individually.
3. The team must submit a brief project report (15-20 pages) that must include the following
   a. Introduction
   b. Requirement Analysis
   c. Software Requirement Specification
   d. Analysis and Design
   e. Implementation
   f. Testing

**Course outcomes:** The students should be able to:

- Design and develop dynamic web pages with good aesthetic sense of designing and latest technical know-how's.
- Understand the concepts of Web Application Terminologies, Internet Tools other web services.
- Recall how to link and publish web sites

**Conduction of Practical Examination:**

1. All laboratory experiments from part A are to be included for practical examination.
2. Mini project has to be evaluated for 40 Marks.
3. Report should be prepared in a standard format prescribed for project work.
4. Students are allowed to pick one experiment from the lot.
5. Strictly follow the instructions as printed on the cover page of answer script.
6. Marks distribution:
   a) Part A: Procedure + Conduction + Viva: **09 + 42 +09 =60 Marks**
   b) Part B: Demonstration + Report + Viva voce **20+14+06 = 40** Marks

Change of experiment is allowed only once and marks allotted to the procedure part to be made zero.

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**INTERNET OF THINGS TECHNOLOGY**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 - 2018)**
**SEMESTER – VIII**

| Subject Code | 17CS81 | IA Marks | 40 |
|---|---|---|---|
| Number of Lecture Hours/Week | 04 | Exam Marks | 60 |
| Total Number of Lecture Hours | 50 | Exam Hours | 03 |

**CREDITS – 04**

| **Module – 1** | **Teaching Hours** |
|---|---|
| What is IoT, Genesis of IoT, IoT and Digitization, IoT Impact, Convergence of IT and IoT, IoT Challenges, 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. | **10 Hours** |
| **Module – 2** | |
| Smart Objects: The "Things" in IoT, Sensors, Actuators, and Smart Objects, Sensor Networks, Connecting Smart Objects, Communications Criteria, IoT Access Technologies. | **10 Hours** |
| **Module – 3** | |
| IP as the IoT Network Layer, The Business Case for IP, The need for Optimization, Optimizing IP for IoT, Profiles and Compliances, Application Protocols for IoT, The Transport Layer, IoT Application Transport Methods. | **10 Hours** |
| **Module – 4** | |
| Data and Analytics for IoT, An Introduction to Data Analytics for IoT, Machine Learning, Big Data Analytics Tools and Technology, Edge Streaming Analytics, Network Analytics, 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 | **10 Hours** |
| **Module – 5** | |
| IoT Physical Devices and Endpoints - Arduino UNO: Introduction to Arduino, Arduino UNO, Installing the Software, Fundamentals of Arduino Programming. IoT Physical Devices and Endpoints - RaspberryPi: Introduction to RaspberryPi, About the RaspberryPi Board: Hardware Layout, Operating Systems on RaspberryPi, Configuring RaspberryPi, Programming RaspberryPi with Python, Wireless Temperature Monitoring System Using Pi, DS18B20 Temperature Sensor, Connecting Raspberry Pi via SSH, Accessing Temperature from DS18B20 sensors, Remote access to RaspberryPi, Smart and Connected Cities, An IoT Strategy for Smarter Cities, Smart City IoT Architecture, Smart City Security Architecture, Smart City Use-Case Examples. | **10 Hours** |

**Course Outcomes:** After studying this course, students will be able to

- Interpret the impact and challenges posed by IoT networks leading to new architectural models.
- Compare and contrast the deployment of smart objects and the technologies to connect them to network.

- Appraise the role of IoT protocols for efficient network communication.
- Elaborate the need for Data Analytics and Security in IoT.
- Illustrate different sensor technologies for sensing real world entities and identify the applications of IoT in Industry.

**Question paper pattern:**

The question paper will have ten questions.
There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

**Text Books:**


1. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry,**"IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things"**, 1st Edition, Pearson Education (Cisco Press Indian Reprint). **(ISBN:** 978-9386873743)
2. Srinivasa K G, **"Internet of Things"**,CENGAGE Leaning India, 2017


**Reference Books:**


1. Vijay Madisetti and ArshdeepBahga, **"Internet of Things (A Hands-on-Approach)"**, 1st Edition, VPT, 2014. **(ISBN:** 978-8173719547)
2. Raj Kamal, **"Internet of Things: Architecture and Design Principles"**, 1st Edition, McGraw Hill Education, 2017. **(ISBN:** 978-9352605224)


H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

## BIG DATA ANALYTICS

**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 - 2018)**
**SEMESTER – VIII**

| | | | |
|---|---|---|---|
| Subject Code | 17CS82 | IA Marks | 40 |
| Number of Lecture Hours/Week | 4 | Exam Marks | 60 |
| Total Number of Lecture Hours | 50 | Exam Hours | 03 |

**CREDITS – 04**

| **Module – 1** | **Teaching Hours** |
|---|---|
| Hadoop Distributed File System Basics, Running Example Programs and Benchmarks, Hadoop MapReduce Framework, MapReduce Programming | **10 Hours** |
| **Module – 2** | |
| Essential Hadoop Tools, Hadoop YARN Applications, Managing Hadoop with Apache Ambari, Basic Hadoop Administration Procedures | **10 Hours** |
| **Module – 3** | |
| Business Intelligence Concepts and Application, Data Warehousing, Data Mining, Data Visualization | **10 Hours** |
| **Module – 4** | |
| Decision Trees, Regression, Artificial Neural Networks, Cluster Analysis, Association Rule Mining | **10 Hours** |
| **Module – 5** | |
| Text Mining, Naïve-Bayes Analysis, Support Vector Machines, Web Mining, Social Network Analysis | **10 Hours** |

**Course outcomes:** The students should be able to:

- Explain the concepts of HDFS and MapReduce framework
- Investigate Hadoop related tools for Big Data Analytics and perform basic Hadoop Administration
- Recognize the role of Business Intelligence, Data warehousing and Visualization in decision making
- Infer the importance of core data mining techniques for data analytics
- Compare and contrast different Text Mining Techniques

**Question paper pattern:**
The question paper will have ten questions.
There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

**Text Books:**

1. Douglas Eadline,**"Hadoop 2 Quick-Start Guide: Learn the Essentials of Big Data Computing in the Apache Hadoop 2 Ecosystem"**, 1st Edition, Pearson Education, 2016. ISBN-13: 978-9332570351
2. Anil Maheshwari, **"Data Analytics"**, 1st Edition, McGraw Hill Education, 2017. ISBN-13: 978-9352604180

**Reference Books:**

1) Tom White, **"Hadoop: The Definitive Guide"**, 4th Edition, O'Reilly Media, 2015.ISBN-13: 978-9352130672
2) Boris Lublinsky, Kevin T.Smith, Alexey Yakubovich,**"Professional Hadoop**

**Solutions",** 1st Edition, Wrox Press, 2014ISBN-13: 978-8126551071
3) Eric Sammer,**"Hadoop Operations: A Guide for Developers and Administrators",**1st Edition, O'Reilly Media, 2012.ISBN-13: 978-9350239261

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**SYSTEM MODELLING AND SIMULATION**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 - 2018)**
**SEMESTER – VIII**

| | | | |
|---|---|---|---|
| Subject Code | 17CS834 | IA Marks | 40 |
| Number of Lecture Hours/Week | 3 | Exam Marks | 60 |
| Total Number of Lecture Hours | 40 | Exam Hours | 03 |

**CREDITS – 03**

| **Module – 1** | **Teaching Hours** |
|---|---|
| **Introduction:** When simulation is the appropriate tool and when it is not appropriate, Advantages and disadvantages of Simulation; Areas of application, Systems and system environment; Components of a system; Discrete and continuous systems, Model of a system; Types of Models, Discrete-Event System Simulation Simulation examples: Simulation of queuing systems. **General Principles, Simulation Software:**Concepts in Discrete-Event Simulation. The Event-Scheduling / Time-Advance Algorithm, Manual simulation Using Event Scheduling | **08 Hours** |
| **Module – 2** | |
| **Statistical Models in Simulation** :Review of terminology and concepts, Useful statistical models,Discrete distributions. Continuous distributions,Poisson process, Empirical distributions.<br>**Queuing Models:**Characteristics of queuing systems,Queuing notation,Long-run measures of performance of queuing systems,Long-run measures of performance of queuing systems cont…,Steady-state behavior of M/G/1 queue, Networks of queues, | **08 Hours** |
| **Module – 3** | |
| **Random-NumberGeneration:**Properties of random numbers; Generation of pseudo-random numbers, Techniques for generating random numbers,Tests for Random Numbers, **Random-Variate Generation:** ,Inverse transform technique Acceptance-Rejection technique. | **08 Hours** |
| **Module – 4** | |
| **Input Modeling:** Data Collection; Identifying the distribution with data, Parameter estimation, Goodness of Fit Tests, Fitting a non-stationary Poisson process, Selecting input models without data, Multivariate and Time-Series input models.<br>**Estimation of Absolute Performance:** Types of simulations with respect to output analysis ,Stochastic nature of output data, Measures of performance and their estimation, **Contd..** | **08 Hours** |
| **Module – 5** | |
| Measures of performance and their estimation,Output analysis for terminating simulations Continued..,Output analysis for steady-state simulations.<br>**Verification, Calibration And Validation:** Optimization: Model building, verification and validation, Verification of simulation models, Verification of simulation models,Calibration and validation of models, Optimization via Simulation. | **08 Hours** |
| **Course outcomes:** The students should be able to: | |

| |
|---|
| • Explain the system concept and apply functional modeling method to model the activities of a static system<br>• Describe the behavior of a dynamic system and create an analogous model for a dynamic system;<br>• Illustrate the operation of a dynamic system and make improvement according to the simulation results. |
| **Question paper pattern:**<br>The question paper will have ten questions.<br>There will be 2 questions from each module.<br>Each question will have questions covering all the topics under a module.<br>The students will have to answer 5 full questions, selecting one full question from each module. |
| **Text Books:** |
| 1. Jerry Banks, John S. Carson II, Barry L. Nelson, David M. Nicol: Discrete-Event System Simulation, 5 th Edition, Pearson Education, 2010. |
| **Reference Books:** |
| 1. Lawrence M. Leemis, Stephen K. Park: Discrete – Event Simulation: A First Course, Pearson Education, 2006.<br>2. Averill M. Law: Simulation Modeling and Analysis, 4 th Edition, Tata McGraw-Hill, 2007 |

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**INTERNSHIP / PROFESSIONAL PRACTISE**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 -2018)**
**SEMESTER – VIII**

| | | | |
|---|---|---|---|
| Subject Code | 17CS84 | IA Marks | 50 |
| Duration | 4 weeks | Exam Marks | 50 |
| | | Exam Hours | 03 |

**CREDITS – 02**

**Description (If any):**

With reference to the above subject, this is to inform that the following are the guidelines to be followed for the Internship Programme and the earlier circular as cited in ref (i) is hereby withdrawn:

1) As per the 15OB.9 the Internship Programme duration is of Eight weeks. However it has been reduced to Four weeks and it should be carried out between (VI and VII Semester) Vacation and/or (VII and VIII Semester) Vacation.

2) The internship can be carried out in any Industry/R and D Organization/Research Institute/ Educational institute of repute.

3) The Institutions may also suggest the students to enrol for the Internshala platform for free internships as there is a MoU with the AICTE for the beneficial of the affiliated Institutions (https://internshala.com/)

4) The Examination of Internship will be carried out in line with the University Project Viva-voce examination.

5) (a) The Department/college shall nominate staff member/s to facilitate, guide and supervise students under internship. (b) The Internal Guide has to visit place of internship at least once during the student's internship.

6) The students shall report the progress of the internship to the guide in regular intervals and seek his/her advice.

7) After the completion of Internship, students shall submit a report with completion and attendance certificates to the Head of the Department with the approval of both internal and external guides.

8) The Examination of Internship will be carried out in line with the University Project Viva-voce examination.

9) There will be 50 marks for CIE (Seminar: 25, Internship report: 25) and 50 marks for Viva – Voce conducted during SEE. The minimum requirement of CIE marks shall be 50% of the maximum marks.

10) The internal guide shall award the marks for seminar and internship report after evaluation. He/she will also be the internal examiner for Viva – Voce conducted during SEE.

11) The external guide from the industry shall be an examiner for the viva voce on Internship. Viva-Voce on internship shall be conducted at the college and the date of Viva-Voce shall be fixed in consultation with the external Guide. The Examiners shall jointly award the Viva - Voce marks.

12) In case the external Guide expresses his inability to conduct viva voce, the Chief Superintendent of the institution shall appoint a senior faculty of the Department to conduct viva-voce along with the internal guide. The same shall be informed in writing to the concerned Chairperson, Board of Examiners (BOE).

13) The students are permitted to carry out the internship anywhere in India or abroad. The University will not provide any kind of financial assistance to any student for carrying out the Internship.

**Course outcomes:** The students should be able to:

1. Adapt easily to the industry environment
2. Take part in team work
3. Make use of modern tools
4. Decide upon project planning and financing.
5. Adapt ethical values.
6. Motivate for lifelong learning

H. O. D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225

**PROJECT WORK PHASE II**
**[As per Choice Based Credit System (CBCS) scheme]**
**(Effective from the academic year 2017 -2018)**
**SEMESTER – VIII**

| | | | |
|---|---|---|---|
| Subject Code | 17CSP85 | IA Marks | 100 |
| Number of Lecture Hours/Week | 06 | Exam Marks | 100 |
| Total Number of Lecture Hours | -- | Exam Hours | 03 |

**CREDITS – 06**

**Description (If any):**

- Project: Carried out at the Institution or at an Industry.
- Project work shall preferably be batch wise, the strength of each batch shall not exceed maximum of four students
- Viva-voce examination in project work shall be conducted batch-wise.
- For Project Phase –I and Project seminar and Project Phase –II, the CIE shall be 100 respectively.
- The CIE marks in the case of projects in the final year shall be based on the evaluation at the end of VIII semester by a committee consisting of the Head of the concerned Department and two senior faculty members of the Department, one of whom shall be the project guide.
- Minimum requirement of CIE marks for Project work shall be 50% of the maximum marks.
- Students failing to secure a minimum of 50% of the CIE marks in Project work shall not be eligible for the Project examination conducted by the University and they shall be considered as failed in that/those Course/s. However, they can appear for University examinations conducted in other Courses of the same semester and backlog Courses if any. Students after satisfying the prescribed minimum CIE marks in the Course/s when offered during subsequent semester shall appear for SEE.
- Improvement of CIE marks shall not be allowed in Project where the student has already secured the minimum required marks
- For a pass in a Project/Viva-voce examination, a student shall secure a minimum of 40% of the maximum marks prescribed for the University Examination. The Minimum Passing Grade in a Course is 'E'.
- The student who desires to reject the results of a semester shall reject performance in all the Courses of the semester, irrespective of whether the student has passed or failed in any Course. However, the rejection of performance of VIII semester project shall not be permitted

**Course outcomes:** The students should be able to:

1. Identify a issue and derive problem related to society, environment, economics, energy and technology
2. Formulate and Analyze the problem and determine the scope of the solution chosen
3. Determine , dissect, and estimate the parameters, required in the solution.
4. Evaluate the solution by considering the standard data / Objective function and by using appropriate performance metrics.
5. Compile the report and take part in present / publishing the finding in a reputed conference / publications
6. Attempt to obtain ownership of the solution / product developed.

H.O.D.
Dept. Of Computer Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225