Basaveshwar Engineering College (Autonomous), Bagalkot

Department of Computer Science and Engineering (CSE)

**Draft Scheme of Syllabus for B. E. (CSE) programme for 175 credits**

Revised Scheme and Syllabus of teaching **(2018-19 Onwards Admitted Batches)**

**Programme: BE (COMPUTER SCIENCE AND ENGINEERING)**

**IIIrd Semester**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Sl. No.** | **Subject Code** | **Subjects** | **Hrs/Week** | | | **C** |  |  |  |
| **L** | **T** | **P** | **CIE** | **SEE** | **Total** |
|  | UMA336C | Computational Methods for Computer Science | 3 | 0 | 0 | 3 | 50 | 50 | 100 |
|  | UCS351C | Digital Systems | 4 | 0 | 0 | 4 | 50 | 50 | 100 |
|  | UCS352C | Computer Organization | 3 | 2 | 0 | 4 | 50 | 50 | 100 |
|  | UCS353C | Object Oriented Programming with Java | 4 | 0 | 0 | 4 | 50 | 50 | 100 |
|  | UCS354H | Professional Communication & Ethics | 2 | 2 | 0 | 3 | 50 | 50 | 100 |
|  | UCS355L | Advance C Programming | 0 | 2 | 2 | 2 | 50 | 50 | 100 |
|  | UCS356L | Digital Systems Lab | 0 | 0 | 2 | 1 | 50 | 50 | 100 |
|  | UCS357L | Object Oriented Programming with Java Lab | 0 | 0 | 2 | 1 | 50 | 50 | 100 |
|  | UMA330M\*\* | Bridge course Maths –I\*\* | 3\*\* | 0 | 0 | 0 | 50 | 50 | 100 |
|  | UBT233M\*\* | Environmental Studies\*\* | 2\*\* | 0 | 0 | 0 | 50 | 50 | 100 |
|  |  |  | **16\*\*** | **6** | **6** | **22** | 500 | 500 | 1000 |

\*\*Note: Diploma lateral entry students have to additionally register the subjects

|  |  |  |
| --- | --- | --- |
| **UMA336C** | **Computational Methods for Computer Science** | **3-CREDITS** |
| **Hrs/Week :3** |  | **CIE Marks:50** |
| **Total Hrs: 48** |  | **SEE Marks:50** |
|  |  |  |

**Course Objectives:**

To enable the students to apply the knowledge of Mathematics in various engineering fields by making them

* to understand the method of solving algebraic, transcendental equations .
* to determine the approximate value of the derivative & definite integral for a given data using numerical techniques.
* able to expand the given periodic function defined in the given range in terms of sine and cosine multiple of terms as a Fourier series.
* able to extremise the functional using integration technique.
* to solve different forms of heat and wave equations.

**Course outcomes**:

On completion of this course, students are able

* to know how root finding techniques can be used to solve practical engineering problems.
* to apply the concept of numerical analysis to find the relative strengths and weaknesses of each computation method and know which are most applicable for given problem.
* to apply the analytical technique to express periodic function as a Fourier sine and cosine series.
* to apply partial differential techniques to solve the physical engineering problems.
* to implement integration technique to determine the extreme values of a functional

**UNIT-I**

**Numerical Analysis-I: 10 Hours**

Introduction to root finding problems, Newton-Raphson method Finite differences, forward and backward difference operators (no derivations on relations between operators). Newton-Gregory forward and backward interpolation formulae (without proof). Lagrange's and Newton's divided difference interpolation formulae (without proof). Numerical differentiation using Newton's forward and backward formulae-problems.

**UNIT-II**

**Numerical Analysis- II: 10 Hours**

Numerical integration: Simpson's one third rule, Simpson's three eighth rule (no derivation ofany formulae)-problems.

**Numerical solutions of ODE & PDE:**

Euler’s and Modified Euler’s method, Runge-Kutta 4th order method .

Numerical solutions of one-dimensional heat and wave equations by explicit method, Laplace equation by using standard five point formula.

**UNIT-III**

**Fourier Series: 10Hours**Periodic functions, Conditions for Fourier series expansions, Fourier series expansion ofcontinuous and functions having finite number ofdiscontinuities, even and odd functions. Half range series, Practical harmonic analysis.

**UNIT-IV**

**Fourier transform: 10 Hours**

Infinite Fourier transforms and inverse Fourier transforms- simple properties,Fourier sine and Fourier cosine transforms.

**Calculusof Variations:**

Variation ofa function and a functional, extremal ofa functional, variational problems, Euler's equation, standard variational problems including geodesics, minimal surface of revolution, hanging chain and Brachistochrone problems.

**Total 40 Hours**

**Resources:**

1. Numerical Methods for Engineers by Steven C Chapra &Raymond P Canale.
2. Higher Engineering Mathematics by Dr. B.S. Grewal, Khanna Publishers, New Delhi.
3. Advanced Engineering Mathematics By H. K. Das, S. Chand & company Ltd. Ram Nagar, New Delhi.
4. Advanced Engineering Mathematics by E Kreyszig ( John Wiley & Sons)

**Question paper pattern for SEE**

1. Total of eight questions with two from each unit to be set uniformly covering the entire syllabus.
2. Each question should not have more than four subdivisions.
3. Any five full questions are to be answered choosing at least one from each unit.

**Assignment Test for 5 Marks:**Ten objective type questions can be prepared from entire syllabus.

|  |  |  |
| --- | --- | --- |
| **UCS351C** | **DIGITAL SYSTEMS** | **4-CREDITS** |
| **Hrs/Week :4** |  | **CIE Marks:50** |
| **Total Hrs: 48** |  | **SEE Marks:50** |
|  |  |  |

|  |  |
| --- | --- |
| **Course Outcomes**  At the end of the course the student should be able to, | |
| CO 1. | Demonstrate the understanding of Boolean algebra. |
| CO 2. | Describe the working of combinational, sequential circuits and Operational Amplifiers and its applications. |
| CO 3. | Apply the Boolean theorems, K-Map, Q-M and VEM methods to simplify Boolean expressions. |
| CO 4. | Design combinational and sequential circuits using MSI digital ICs. |
| CO 5. | Simulate combinational circuits using HDL programming. |

|  |
| --- |
| **UNIT-I** (**12 Hours)**  **Boolean algebra and Combinational Circuits:** Boolean algebra definition, principle of duality, Boolean algebra theorems, Boolean formulas and functions, normal forms. Minterm canonical form, m-notation, Maxterm Canonical form, M-notation. Manipulation of Boolean expressions. Gates and combinational circuits. Incomplete Boolean functions and don’t care conditions, Additional Boolean operations and Gates, Introduction to HDL. |
| **UNIT-II (12 Hours)**  **Simplification of Boolean expressions**: Karnaugh-maps, Use of Karnaugh-maps to minimize Boolean Expressions. Minimal Expressions of Incomplete Boolean Functions. The Quine-McCluskey and Decimal methods for generating prime implicants and prime implicates. Map Entered Variables(MEV). |
| **UNIT-III** (**12 Hours)**  **Logic Design using MSI Components:** Binary Adders and Substractor, Comparators, Decoders, Encoders, Multiplexers.  **Flip Flops and its Applications:** Basic bistable element, Latches, Master Slave SR and JK flip-flops, Edge Triggered flip-flops, Characteristic Equations. HDL implementations of logic circuits. |
| **UNIT-IV** (**12 Hours)**  **Registers**, **Counters**, Design of synchronous counters. HDL implementation of flip-flop, registers and counters.  **Operational Amplifiers and its Applications:** Introduction to operational amplifiers., Block diagram representation of a typical Op-Amp, Equivalent Circuits of an Op-Amps, Ideal Voltage Transfer curve, Open Loop Op-Amp Configurations, Digital to Analog –Analog to Digital conversion using Op-Amps: |

|  |
| --- |
| **Text Books:**   1. D.D. Givone, 2002, **‘Digital Principles and design’**, McGraw Hill. 2. Ramakant A. Gayakwad, 2008, **‘Op-Amps and Linear Integrated Circuits’**, 4th Edition |
| **Reference Books:**   1. Malvino, Leach and Saha **‘Digital Principles and applications’**, 6th Edition, 2007, McGraw Hill. 2. R. D. Sudhakar Samuel, **“Logic Design - A simplified approach”** Revised Edition, 2005, Sanguine Technical Publications. 3. Stephen Brown & Zvonko Vranesic, **“Fundamental of digital Logic with Verilog Design”**Publication Tata McGraw Hill. |

|  |  |  |
| --- | --- | --- |
| **UCS352C** | **COMPUTER ORGANIZATION** | **4-CREDITS** |
| **Hrs/Week:04 L:T:P:3:2:0** | | **CIE Marks:50** |
| **Total Hrs:40:24:0** |  | **SEE Marks:50** |

|  |  |
| --- | --- |
|  | **Course Outcomes**  At the end of the course the student should be able to |
| CO 1. | Explain the design and function of different units of single and multiprocessors system |
| CO 2. | Analyze the execution of the program and different organizations of functional units |
| CO 3. | Compare the performance of single and multiprocessor systems |
| CO 4. | Develop an assembly programs and micro programs for simple machine instructions |
| CO 5. | Design the basic functional units of computer |

|  |
| --- |
| **Unit I (10 + 6) Hours**  **Basic structure of Computers:** Computer types, Functional Units, Basic operational concepts, Bus structures  **Machine instructions and programs:** Numbers, Arithmetic operations and characters, Memory locations and addresses, Memory operations, Instructions and instruction sequencing, Addressing modes, Assembly language, assembler directives, number notation, , Stacks and Queues, Subroutines, Encoding of machine instructions |
| **Unit II (10 + 6) Hours**  **Input/output organization:** Accessing I/O devices, Interrupts-Interrupt hardware , Enabling and Disabling Interrupts, Handling Multiple devices, controlling device requests, Exceptions, Direct memory access – Bus Arbitrations, Buses- Asynchronous Bus and Synchronous bus , Interface Circuits- Parallel port and serial port, Standard I/O Interfaces –Peripheral component interconnect Bus, SCSI bus, USB. |
| **Unit III (10 + 6) Hours**  **The memory system:** Some Basic concepts, Semiconductor RAM memories, Read only memories, speed, size, and cost, cache memories  **Arithmetic Unit:** 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 – IEEE standard for Floating point numbers, Arithmetic operations on Floating point numbers. Implementing Floating point operations. |
| **Unit IV (10 + 6) Hours**  **Basic Processing Unit:** Some fundamental concepts, Execution of complete instruction, Hardwired Control, Micro programmed control, Microinstructions,  **Pipelining:** basic concepts, role of cache memory, pipeline performance  **Large computer systems**: forms of parallel processing, array processor, the structure of general purpose and multiprocessors  **Performance:**  Processor Clock, Basic performance equation, pipelining and superscalar operations, Clock rate, Instruction set, compiler, performance measurement. |
| **Text Books:**  1. Hamacher, Zvonko Vranesic, Safwat Zaky, 2002. ‘Computer Organization’, Fifth Edition, MGH.  (1.1 to 1.4, 2.1 to 2.5, 2.6.1, 2.6.3, 2.8 to 2.9, 2.12, 4.1, 4.2,4.2.1 to 4.2.5, 4.4, 4.4.1,4.5,4.5.1 to 4.5.2, 4.6 , 4.7, 5.1 to 5.5, 5.5.1, 6.1 to 6.7, 7.1 to 7.5, 7.5.1, 8.1,8.1.1,8.1.2,12.1-12.3, 1.6) |
| **Reference Book:**   1. J.P. Hayes, 1998, ’Computer Architecture and Organization ‘ , Third Edition, MGH.   2. William Stallings, 2007 ‘Computer Organization and Architecture’, 7th Edition, PHI. |

|  |  |  |  |
| --- | --- | --- | --- |
| **UCS353C** | | **Object Oriented Programming with Java** | **4-CREDITS** |
| **Hrs/Week :(4)** | | L:T:P (4+0+0) | **CIE Marks:50** |
| **Total Hrs:48** | |  | **SEE Marks:50** |
| **Course Outcomes**  At the end of the course the student should be able to , | | | |
| CO 1. | Explain the object-oriented concepts and other features of JAVA. | | |
| CO 2. | Identify classes and relationship among them needed for the problem given. | | |
| CO 3. | Design and develop standalone applications using Java | | |
|  |  | | |
|  |  | | |
| **Unit I (12 Hours)**  **Java Programming Fundamentals**: Object Oriented programming features  **History and evolution of Java**: Java's lineage, bytecode, Java Buzzwords**.**  **An overview of Java** ,**Data Types, Variables and Arrays** , **Operators , Control Statements,**  **Introducing Classes: Class Fundamentals** , Declaring Objects , Introducing Methods , Constructors ,this keyword ,garbage collection, method overloading. | | | |
| **Unit II ( 12 Hours)**  **Inheritance ,Packages and Interfaces**  **String Handling ,Enumerations, Autboxing and Type wrappers, Exception Handling :** Exception-Handling Fundamentals – Exception Classes , Exception Types, Uncaught Exceptions, Using try and catch, Multiple catch clauses, Nested try Statements, throw and finally statements. | | | |
| **Unit III ( 12 Hours)**  **Lambda Expressions :** Fundamentals, Block Lambda expressions, Passing Lambda Expressions as argument, Lambda Expressions and Exceptions ,Method References.  **Multithreaded Programming :** The Java Thread Model , The Main Thread , Creating a Thread, Creating Multiple Threads, Using isAlive( ) and join() , Thread Priorities , Synchronization , Suspending, Resuming and Stopping Threads | | | |
| **Unit IV ( 12 Hours)**  **Files** : Byte streams, Character Streams, Serialization, Console Class. **Regular Expressions**: Regular Expressions Processing.  **Collections :**The Collection Interfaces , The Collection Classes(ArrayList , LinkedList) , Accessing a Collection via an Iterator , Legacy Classes and Interface. | | | |
| **Text Book:**  Java The Complete Reference - Herbert Schildt 9th Edition, MGH Education  **Reference Books:**  1.Core Java Volume 1- Fundamentals, Cay S Horstmann ,Gary Cornell, 8th Edition ,Pearson Education  2.Programming with Java – E Balagurusamy,6th Edition, MGH | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **UCS355L** | | **Advance C Programming** | **2-CREDITS** |
| **Hrs/Week : 4(2+2)** | | **L:T:P:0:2:2** | **CIE Marks:50** |
| **Total Hours:28** | |  | **SEE Marks:50** |
|  | |  |  |
|  | **Course Outcomes**  At the end of the course the student should be able to , | | | |
| CO 1. | Explain the C programming language concepts. | | | |
| CO 2. | Analyze and determine the program requirements | | | |
| CO 3. | Employ the concepts in designing and developing effective solutions for the problem given | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Unit I (07 Hours)  Multidimensional arrays, String handling functions , Command line arguments, **preprocessor commands:** file inclusion, macro definition, conditional compilation  **Storage classes:** Storage classes**,** object storage attributes, storage class specifiers,  **Type qualifiers:** constant, volatile and restricted type qualifiers. | | | |
| Unit II (07Hours)  **Pointers:** Introduction, pointers for Inter- function communication, pointers to pointers, compatibility,  **Pointer applications**: Arrays and pointers, pointer arithmetic and arrays, passing an array to a function,  **Memory allocation functions, Array of pointers, pointers to void and pointers to functions Recursion:** iterative and recursive definitioniterative and recursive solution, designing recursive functions, limitations of recursion. | | | |
| Unit III (07 Hours)  **Enumerators, Structures and Union Types:** typedef **,** enumerated types, structure, unions  **Bitwise operators**: Exactsize integer types, logical bitwise operators, shift operators, masks  **Variable argument list functions,** | | | |
| Unit IV (07 Hours)  **Files: Text Input/output** :files, streams, standard library I/O functions, formatting I/O functions, character I/O functions  **Binary files**: Text v/s binary stream, standard library function for files, converting file types | | | |
| **Text Book:**  Computer Science A Structured Programming Approach Using C, Behrouz A,Forouzan & Richard F Gilberg, Third Edition, Cengage Learning India Private Limited (Chapter 6:6.9 Chapter 7,Chapter 9 &10,Chapter 11:11.3,11.4,11.5,Chapter 12,Chapter 13,Chapter 14,Appendix G:G.1,G.2,G.3,Appendix H,I and J) | | | |
| **Reference Books:**   1. Let Us C ,Yeshwant Kanetkar,15th Edition, BPB publications 2. Programming with C ,Brian W Kernighan and Dennis Ritchie ,Pearson Education 3. Test your C Skills , Yeshwant Kanetkar, BPB publications 4. Exploring C Yeshwant Kanetkar BPB publications | | | |
|  | | | |
| **Assignment Set** | | | |
| **Part A:**   1. Program to create 3d array regular(rectangular) array 2. Program to create dynamic 2d array of user specified size 3. Program on command line arguments and use of macros 4. Program to implement simple string library functions 5. Develop program to perform the following using recursive function   i) Print the numbers accepted from keyboard in reverse order  ii) Obtain sum of array elements, even elements, odd elements  iii) Obtain length of array, occurrence count, reverse array elements  iv) Reverse the number; sum the digits of a number, binary to decimal   1. Demonstrate writing and using of variable argument list function 2. Demonstrate nested structure concept, array of structures, array of pointers to structure with dynamic allocation for array elements 3. Program to perform following using bitwise operators on fixed size integers of 8 bit length   i)display number in 1’s and 0’s  ii)use of bitwise or,and ,not and exclusive or operator  iii)masking ,unmasking,flipping,left shifting ,right shifting,rotation   1. Program to create ,display text file and separate the words and store in another file 2. Merge two files containing integers (that may not be sorted ) such that resultant file is sorted 3. Program to create binary file of students and allow following operations   i)To display in forward and reverse direction  ii)To modify student record given key value  iii)To retrieve nth record  **Part B: For practice purpose only**  1)Debugging exercises  2)Determining the output for given piece of core  3)Finding error in the code given  4)Complete the code  5)Given prototype develop the function  6)Given the code identify the task. | | | |
| **Note:**   * In the examination questions must be given on lots. Each student must be given one question only from PART-A. | | | |
| UCS354H | | | **Professional Communication and Ethics** | **3 credits (2-2-0)** | |
| **Hrs/Week : 4(2+2)** | | | **L:T:P:2:2:0** | **CIE Marks:50** | |
| **Total Hours: 28L+24T** | | |  | **SEE Marks:50** | |
|  | | |  |  | |
|  | | **Course Outcomes :**  At the end of the course the student should be able to | | | | |
| CO 1. | | Apply communication skills effectively in profession and society. | | | | |
| CO 2. | | Recognize the importance of communication, listening, team work and behavior. | | | | |
| CO 3. | | Execute ethical and social responsibility as an IT professional. | | | | |
| CO 4. | | Express ideas to produce messages suitably tailored for the topic, objective, audience, communication medium and context. | | | | |
| CO 5 | | Understand corporate culture and perform accordingly. | | | | |

|  |
| --- |
| **UNIT I L – 7 Hrs T- 6 Hrs**  **Basics of Technical Communication:** Introduction, Process of Communication, Levels of Communication, Flow of Communication, Visual aids in Technical Communication.**Barriers to Communication:** Introduction, Classification of Barriers, **Non-verbal Communication:** Introduction, Kinesics, Proxemics, Chronemics, Correlating Verbal and Non-verbal Communication, Cross-cultural variations, |
| **UNIT II L – 7 Hrs T- 6 Hrs**  **Active Listening:** Introduction, Types of Listening, Traits of Good Listener, Active Versus Passive Listener, Implication of Effective Listening, **Conversations and Dialogues:** Introduction, Conversations, Telephonic conversations and Etiquettes, Dialogue writing **Formal Presentations:** Introduction, Planning, Outlining and Structuring, Nuances of Delivery, Controlling Nervousness and Stage, Fright, Visual Aids in Presentation, Application of MS-PowerPoint |
| **UNIT III L – 7 Hrs T- 6 Hrs**  **Group Communication:** Introduction, Forms of Group Communication, Use of Body Language, Discussions, Group Discussions, Organizational Group Discussion, Group Discussion as a Part of Selection Process Meeting, Conferences, **Elements of Effective Writing:** Introduction, Right Words and Phrases, Sentences, Writing for the Web |
| **UNIT IV L – 7 Hrs T- 6 Hrs**  **Ethics:** An Overview of Ethics, What Are Ethics? Ethics in the Business World, Including Ethics Considerations in Decision Making, Ethics in Information Technology, **Ethics for IT Workers and IT Users:** IT Professionals, IT Users **Software Development:** Strategies of Engineering Quality Software, Software Product Liability, Key Issues in Software Development,  **Social Networking:** What is a Social Networking Web Site? Social Networking Ethical Issues |

|  |
| --- |
| **Text Books:**  **Communication:**   1. Meenakshi Raman And Sangeeta Sharma –Technical Communication-Principles and   Practice, Oxford University Press ,2004(Chapters 1, 2, 3, 4, 6, 7, 9 and 11)  **Ethics:**  2. George Reynolds –Ethics in Information Technology , Thomson Course  Technology,2003(Chapters 1, 2, 7 and 9) |
| **Reference Books:**   1. M. Ashraf Rizivi, Effective Technical Communication,2nd Edition, McGraw Hill, 2017 2. Mike W Martin And Ronald Schinzinger, Ethics in Engineering, 2nd Edition, McGraw Hill, 2010 3. Aruna Koneru, Professional Communication, Tata McGraw-Hill Education, 2008 4. Jayashree Suresh and B. C. Raghavan, Human Values and Professional Ethics, S. Chand and Company, 2010 |
| **Question paper pattern:**   * Total of 8 questions, two from each unit, to be set uniformly covering the entire syllabus. * No question should have more than 4 sub divisions. * Any five full questions are to be answered choosing at least one from each unit. * The questions should cover all the levels of Bloom’s taxonomy. * All COs have to be covered. |

|  |  |  |
| --- | --- | --- |
|  | **DIGITAL SYSTEMS LABORATORY** |  |
| **UCS356L**  **Hours/Week: 02** |  | **1-CREDITS**  **CIE Marks: 50** |
| **Exam Hours:02** |  | **SEE Marks: 50** |

|  |  |
| --- | --- |
|  | **Course Outcomes**  At the end of the course the student should be able to , |
| CO 1. | This course provides the foundation education in digital electronic circuit analysis and design. Through lecture, laboratory, and out-of-class assignments, students are provided learning experiences that enable them to: |
| CO 2. | Design, simulate and implement basic combinational and sequential logic circuits. Become proficient with computer skills (eg., VHDL language) for the analysis and design of circuits. |
| CO3 | Acquire teamwork skills for working effectively in groups . |
|  |  |

|  |
| --- |
| **Practice Assignments using digital I C’s :**   * Implementation of Boolean Expressions of basic logic gates such as 2-input/3-input AND,OR,NAND,NOR, EX-OR gates * Simplification of simple Boolean Expressions in SOP/POS forms. |
| **PART- A (Hardware Implementation)**   1. Design a Binary to Gray Code converter with K map simplification and using basic Gates. 2. Given any 4-variable logic expression, simplify using K-MAP/Quine McCliskey and realize the simplified logic expression using 8:1 multiplexer IC. 3. Realize a full adder using 3-to-8 decoder IC and 4 input NAND gates. 4. Realize a full substractor circuit using 3 to 8 line decoder IC and 4 input NAND gate. 5. Realize a J-K Master/Slave Flip-Flop using NAND gates and verify its truth table. 6. Design and implement a mod-n (n<8) synchronous Up Counter using J-K Flip-Flop and basic gate ICs. 7. Design and implement a mod-n (n<8) synchronous Down Counter using J-K Flip-Flop and basic gate ICs. 8. Design and implement an asynchronous counter using decade counter IC to count up from 0 to n (n<=9) & display the numbers using 7-segment display. 9. Design a Ring and Johnson Counter using a 4-bit Shift Register IC. 10. Design a 4-bit R-2R ladder D/A converter using Op-Amp. Determine its accuracy and resolution.   **Practice Assignments using Simulation package :**   * Implementation of Boolean Expressions of basic logic gates such as 2-input/3-input AND,OR,NAND,NOR, EX-OR gates * Simplification of simple Boolean Expressions in SOP/POS forms. |
| **PART- B (Software Implementation)**   1. Write the Verilog/VHDL code for Binary to Gray Code converter and verify its working. 2. Write the Verilog/VHDL code for an 8:1 multiplexer. Simulate and verify its working. 3. Write the verilog/VHDL code for a full adder .Simulate and verify its working. 4. Write the Verilog/VHDL code for D Flip-Flop with positive-edge triggering. Simulate and verify its working. 5. Write a verilog/VHDL code for mod-8 up counter. Simulate and verify its working. 6. Write the verilog/VHDL code for switched tail counter. Simulate and verify its working.     **Note:**   * Any simulation package like MultiSim/Active HDL etc. may be used. * In the examination questions must be given on lots. Each student must be given one question from PART-A and one from PART-B. * Practice Assignments are not to be considered for SEE Examination. |

|  |  |  |
| --- | --- | --- |
| **UCS357L** | **Object Oriented Programming with Java Lab** | **1-CREDIT** |
| **Hrs/Week :02** |  | **CIE Marks:50** |
| **Exam Hrs:03** |  | **SEE Marks:50** |

|  |  |
| --- | --- |
|  | **Course Outcomes**  At the end of the course the student should be able to , |
| CO 1. | Analyze the problem statement and determine the requirements for solving problem. |
| CO 2. | Design and develop effective solution for the problem given. |
|  |  |
|  |  |

|  |
| --- |
| 1. Develop simple applications demonstrating the 2. Use of conditional statements   ii) Use of loop statements  iii) Reading & printing different data types in java  iv) operations on arrays(single & multidimensional)   1. Develop application demonstrating 2. Inheritance 3. Polymorphism 4. Packages 5. interfaces   3. Develop applications demonstrating exception handling  4. Develop applications demonstrating multithreading concept  i) Creating threads using extends & runnable technique  ii) Synchronization  iii) Interthread Communication  5. Develop application that demonstrates the use of  i) String library functions  6. Develop applications that allows manipulation of collections  i) ArrayList ii)Linked List  7.Develop application that allows user to create a file,display and manipulate file  i) using byte stream  ii) character stream   1. using object stream |

**Programme: BE (COMPUTER SCIENCE AND ENGINEERING)**

**IVth Semester**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Sl. No.** | **Subject Code** | **Subjects** | **Hrs/Week** | | | **C** |  |  |  |
| **L** | **T** | **P** | **CIE** | **SEE** | **Total** |
|  | UMA436C | Statistics and Probability Theory | 3 | 0 | 0 | **3** | 50 | 50 | 100 |
|  | UCS451C | Data Structures using C | 3 | 2 | 0 | **4** | 50 | 50 | 100 |
|  | UCS452C | Database Management Systems | 4 | 0 | 0 | **4** | 50 | 50 | 100 |
|  | UCS453C | Operating Systems | 3 | 2 | 0 | **4** | 50 | 50 | 100 |
|  | UCS454C | Microcontrollers | 3 | 0 | 0 | **3** | 50 | 50 | 100 |
|  | UCS455L | Data Structures using C Lab | 0 | 0 | 2 | 1 | 50 | 50 | 100 |
|  | UCS456L | Microcontrollers Lab | 0 | 0 | 2 | **1** | 50 | 50 | 100 |
|  | UCS457L | Database Management Systems Lab | 0 | 0 | 2 | **1** | 50 | 50 | 100 |
|  | UHS001N | Fundamentals Of Quantitative Aptitude And Soft Skills | 0 | 2 | 0 | **1** | 50 | 50 | 100 |
|  | UMA430M\*\* | Bridge course Maths –II\*\* | 3\*\* | 0 | 0 | **0** | 50 | 50 | 100 |
|  | UHS126M\*\* | Constitution of India\*\* | 2\*\* | 0 | 0 | **0** | 50 | 50 | 100 |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  | **16\*\*** | **6** | **6** | **22** | **600** | **600** | **1200** |

Bridge course Maths II is mandatory subject only for students having Diploma and admitted to 4th Semester through lateral entry scheme. Passing the subject is compulsory; however marks will not be considered for awarding grade/class. A PP/NP grade will be awarded for passing /not passing the subject.

* The total lecture hours for students having Diploma and admitted to 4th Semester through lateral scheme is 26 hours.

Note: Diploma lateral entry students have to additionally register for CIP.

|  |  |  |
| --- | --- | --- |
| **UCS436C** | **Statistics and Probability Theory** |  |
| **Hrs/Week : 03** | **L:T:P:3:2:0** | **CIE Marks:50** |
| **Total Hrs: 40** |  | **SEE Marks:50** |

**Course Objectives:**

To enable the students to apply the knowledge of Mathematics in various Engineering fields by making them

* To form a specific relation for the given group of data using least square sense method.
* To specify probability is an area of study which involves predicting the relative likely hood of various outcomes.

**Course outcomes**:

On completion of this course, students are able

CO1: to apply the least square sense method to construct the specific relation for the given group of data.

CO2: to apply the concept of probability to find the physical significance of various distribution phenomena.

CO3: to apply the concept of probability to perform engineering duties in planning and designing, engines, machines and other mechanically functioning.

CO4: to apply the concept of probability to study the performance of Mechanical systems.

CO5: to apply the concept of Markov Chain for commercial and industry purpose.

**UNIT-I**

**Statistics: 10 Hours**

Curve fitting by the method of least squares :.

Correlation , expression for the rank correlation coefficient and regression.

**UNIT –II**

**Probability: 10 Hours**

Probability: Addition rule, conditional probability, multiplication rule, Baye’s rule. Discrete and continuous random variables-Probability density function, cumulative distribution function.

**UNIT –III**

**Probability distributions: 10 Hours**

Binomial distribution, Poisson distribution and Normal distribution. Concept of joint probability, Joint distributions - discrete and continuous random variables, Independent random variables, problems on expectation and variance.

**UNIT –IV**

**Markov chains: 10 Hours**

Markov chains: Introduction, Probability vectors, Stochastic Matrices, Fixed Points and Regular stochastic Matrices, Markov chains, higher transition probabilities, stationary distribution of regular Markov chains and absorbing states.

**Total: 40 Hours**

**Resources:**

1. Higher Engineering Mathematics by Dr. B.S. Grewal, Khanna Publishers, New Delhi.
2. Theory and problems of probability by Seymour Lipschutz (Schaum’s Series).
3. Advanced Engineering Mathematics by H. K. Dass
4. Advanced Engineering Mathematics by E Kreyszig ( John Wiley & Sons)
5. Probability and stochastic processes by Roy D. Yates and David J. Goodman, wiley India pvt.ltd 2nd edition 2012.
6. Advanced Engineering Mathematics by Peter V. O’Neil.

**Question paper pattern for SEE**:

1. Total of eight questions with two from each unit to be set uniformly covering the entire syllabus.
2. Each question should not have more than four subdivisions.
3. Any five full questions are to be answered choosing at least one from each unit.

**Assignment Test for 5 Marks:** Ten objective type questions can be prepared from entire

syllabus.

|  |  |  |
| --- | --- | --- |
| **UCS451C** | **DATA STRUCTURE USING C** | **4-CREDITS** |
| **Hrs/Week : (03+02)** | **L:T:P:3:2:0** | **CIE Marks:50** |
| **Total Hrs: 52** |  | **SEE Marks:50** |

|  |  |
| --- | --- |
|  | **Course Outcomes**  At the end of the course the student should be able to , |
| CO1. | Explain linear and nonlinear data structures concepts, searching and sorting techniques. |
| CO2. | Analyze and implement different data structures, searching and sorting techniques. |
| CO3. | Develop solutions for the given problem by using relevant data structures. |

|  |
| --- |
| Unit I (13 Hours)  **Introduction to Data Structures : Basic concepts :**Abstract data type: Atomic and composite data, Data type, Data structure, Abstract data type, Model for an abstract data type: ADT operations, ADT data structures, Pointer to void, **Pointer to Function**: Defining pointers to functions, Using pointers to functions.  **Stacks**: **Basic stack operations**: Push, Pop, Stack top, **Stack linked list**:Implementation, Data structure, Stack head, Stack data node, Stack algorithms, Create Stack, Push Stack, Stack top, Empty Stack, Full Stack, Stack count, Destroy Stack, **C language implantations**: Insert data, Push Stack , Print Stack, Pop character, **Stack ADT**: Data structure, ADT Implementations, Stack structure, Create stack, Push stack, Pop stack, Stack top, Empty stack, Stack count, Destroy stack, **Stack applications**: Reversing data, Reverse a list, Convert decimal to binary, Infix to postfix transformation, Evaluating postfix expressions, Stack Implementation using array. |
| Unit II (13 Hours)  **Queues**: **Queue Operations**: Enqueue, Dequeue, Queue front, Queue rear, Queue example, **Queue Linked list design**: Data structure, Queue head, Queue data node, Queue algorithms, Create queue, Enqueue, Dequeue, Retrieving queue data, Empty queue, Full queue, Queue count, Destroy queue, **Queue ADT**: Queue structure, Queue ADT algorithms, Queue Implementation using array, Queue Applications.  **Sorting** :Selection, Insertion, exchange and quick sorts  **Searching**: Sequential, binary search, hashed list searches |
| Unit III (13 Hours)  **General Linear lists: Basic operations**, Insertion, Deletion, Retrieval, Traversal, **Implementation:** Data structure, Head node, Data node, Algorithms, Create list, Insert node, Delete node, List search, Retrieve node, Empty list, Full list, List count, Traverse list, Destroy list, **List ADT**: ADT functions, Create list, Add node, Internal insertion function, Remove node, Internal delete function, Search list, Internal search function, Retrieve node, Empty list Full list, List count, Traverse, Destroy list,  **Circular linked lists and Doubly linked lists**: Create list, add node, delete node, retrieve node, search list. |
| Unit IV ( 13 Hours)  **Non-Linear lists: Trees: Basic tree concepts**: Terminology, User representation, **Binary trees**: Properties, Height of binary trees, Balance, Complete and Nearly complete binary trees, **Binary tree traversals:** Depth-first traversals, Breadth-first traversals, **Expression Trees**:Infix traversal, Postfix traversal, Prefix traversal, **Huffman code, General trees, Binary search trees**: Basic concepts, **BST operations**: Traversals, Searches, Insertion Find the smallest and largest node, BST search, Insertion, Deletion, **Binary search tree ADT**, Data structure, Head and node structure, Algorithms, Create a BST, Insert a BST, Internal insert function, Delete a BST, Internal delete function, Retrieve a BST, Internal retrieve function, Traverse a BST, Internal traverse function, Empty a BST, Full BST, BST count, Destroy a BST, Internal destroy function.  **Graphs: Basic concepts**, **Operations**: Insert vertex, Delete vertex, Add edge, Delete edge, Find vertex, **Graph storage structures**: Adjacency matrix, Adjacency list. |
| **Text Book:**   1. Behrouz A. Forouzan and Richard F. Gilberg, 2nd Edition, Cengage Learning Publisher, 2005. **Data Structure A Pseudocode Approach with C**, (Chapter 1(1.2,1.3,1.5), 2,3,4 (4.1-4.4), 5, 6(6.1-6.3 )7(7.1-7.3), 11(11.1-11.3),12(12.2-12.4)13(13.1-13.3)Appendix F. |
| **Reference Books:**   1. **Data Structures Using C,** Aaron M. Tenanbaum , Yedidyah Langsam, Moshe J Augenstein Pearson Education 2. Data **Structures and Program Design in C,** Robert Kruse, Bruce Leung, C. L. Tondo , Shashi Mogalla, 2nd Edition, Pearson Education 3. **Data Structures with C ,Seymour Lipschutz, Schaum’s outlines, MGH Education** 4. **Data Structures Through C ,Yeshwant Kanetkar ,BPB publications** |

|  |  |  |
| --- | --- | --- |
| **UCS452C** | **DATABASE MANAGEMENT SYSTEMS** | **4-CREDITS** |
| **Hrs/Week :04** |  | **CIE Marks:50** |
| **Total Hrs: 48** |  | **SEE Marks:50** |

|  |  |
| --- | --- |
|  | **Course Outcomes**  At the end of the course the student should be able to: |
| CO 1. | Explain the concepts of database and database management system. |
| CO 2. | Describe security concepts for multi user database applications. |
| CO 3. | Design database for given database application. |
| CO 4. | Apply normalization concepts to refine designed database. |
| CO 5. | Develop database programming skills. |

|  |
| --- |
| **Unit I ( Hours)**  **Introduction and Entity-Relationship Model** **L - 12 Hrs**  Introduction; Characteristics of Database approach; People with databases; Advantages; Disadvantages of DBMS. Data models, schemas and instances; Three-schema architecture and data independence; Database languages and interfaces; The database system environment; Classification of Database Management systems. Using High-Level Conceptual Data Models for Database Design; An Example Database Application; Entity Types, Entity Sets, Attributes and Keys; Relationship types, Relationship Sets, Roles and Structural Constraints; Weak Entity Types; Refining the ER Design for COMPANY database; ER Diagrams, Naming Conventions and Design Issues; Relationship types of degree higher than two. |
| Unit II ( Hours)  **Relational Model and Relational Algebra L - 12 Hrs**  Relational Model Concepts; Relational Model Constraints and Relational Database Schemas; Update Operations, Transactions and dealing with constraint violations; Unary Relational Operations: SELECT and PROJECT; Relational Algebra Operations from Set Theory; Binary Relational Operations : JOIN and DIVISION; Additional Relational Operations; Examples of Queries in Relational Algebra; Relational Database Design using ER- to-Relational Mapping.  **SQL-The Relational Database Standard:**  SQL Data Definition and Data Types, Specifying Basic Constraints in SQL, Schema Change Statements in SQL; Basic Queries in SQL; More Complex SQL Queries; Additional Features of SQL; Views (Virtual Tables) in SQL, Cursors, Triggers and PL/SQL Programming |
| Unit III ( Hours)  **Database Design L - 12 Hrs**  Informal Design Guidelines for Relation Schemas; Functional Dependencies; Normal Forms Based on Primary Keys; General Definitions of Second and Third Normal Forms; Boyce-Codd Normal Form; Properties of Relational Decompositions; Algorithms for Relational Database Schema Design; Multivalued Dependencies and Fourth Normal Form; Join Dependencies and Fifth Normal Form; |

|  |
| --- |
| Unit IV ( Hours)  **Transaction Management and Recovery Techniques L - 12 Hrs**  Introduction to transaction processing; Transaction and System concepts; The ACID Properties; Characterizing Schedules Based on Recoverability; Two-Phase Locking Technique for concurrency Control(2PL); Recovery Concepts; Recovery and backup Techniques Based on Deferred Update and Immediate Update |
| **Text Books:**   1. Elmasri and Navathe,2007, **’Fundamentals of Database Systems’**, 5th Edition, Addison-Wesley, 2007 |
| **Reference Book:**   1. Silberschatz, Korth and Sudharshan: 2006,’ **Data base System Concepts’**, 5th Edition, Mc-GrawHill 2. **‘Database Management Systems’**, Raghurama Krishnan, Johannes Gehrke, TATA Mc-GrawHill |
|  |

|  |  |  |
| --- | --- | --- |
| **UCS453C** | **Operating systems** | **4-CREDITS** |
| **Hrs/Week :04** | **(L:T:P:S):03:02:00** | **CIE Marks:50** |
| **Total Hrs:52** |  | **SEE Marks:50** |

|  |  |
| --- | --- |
|  | **Course Outcomes**  At the end of the course the student should be able to , |
| CO 1. | List and explain goals, service and functions of different classes of operating systems |
| CO 2. | Analyze the performances of different process scheduling, memory management, file system implementation, protection and security mechanisms. |
| CO 3. | Apply scheduling and memory allocation policies for solving scheduling and memory management problems. |
| CO 4. | Develop simple concurrent applications using processes and threads |
| CO 5. | Select appropriate mechanisms for deadlock handling, synchronization and interprocess communication. |

|  |
| --- |
| Unit I ( 10+6 Hours)  **Introduction** : Abstract Views of an Operating System , Goals of an Operating System , Operation of an Operating System, Operating System and the Computer System, Efficiency, System Performance and User Convenience, Classes of Operating Systems, Batch Processing Systems, Multiprogramming Systems, Time Sharing Systems, Real Time Operating Systems, Distributed Operating Systems , Modern Operating Systems  **Processes and Threads**: Processes and Programs, Programmer View of Processes , Operating System View of Processes. Threads,  **Scheduling :** Preliminaries, Non-preemptive Scheduling Policies, Preemptive Scheduling Policies, Scheduling in Practice , Real Time Scheduling. |
| Unit II ( 10+6 Hours)  **Synchronisation :** Background, The critical section problem,Petersons solution, Synchronisation hardware, Semaphores, Classic Problems of synchronization, Monitors.  **Deadlocks:** System model, Deadlock Characterization, Methods of Handling deadlocks ,Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery From Deadlock |
| Unit III ( 10+6 Hours)  **Memory Management :** Managing the Memory Hierarchy, Static and Dynamic Memory Allocation, Memory Allocation to a Process, Reuse of Memory, Contiguous Memory Allocation, Noncontiguous Memory Allocation, Paging, Segmentation, Segmentation with Paging.  **Virtual Memory :** Virtual Memory Basics, Demand Paging , Page Replacement Policies, Memory Allocation to a Process, Shared Pages, Memory Mapped Files. |
| Unit IV ( 10+6 Hours)  **File Systems:** File System and Input Output control system(IOCS), Files and File Operations, Fundamental File Organizations, Directory Structures, File Protection, Interface between File System and IOCS, Allocation of Disk Space, Implementing File Access, File Sharing Semantics, File System Reliability, Virtual File System .  **Security and Protection :** Overview of Security and Protection, Goals of Security and Protection, Security Attacks, Formal and Practical Aspects of Security, Encryption, Authentication and Password Security, Access Descriptors and the Access Control Matrix, Protection Structures, Capabilities.  **Case Study :** The Linux System: Linux History, Design Principles, Kernel Modules, Process Management, Scheduling, Memory Management, File Systems, Input and Output,Interprocess Communication, Security. |
| **Text Books:**   1. D. M. Dhamdhere, Operating Systems--A Concept Based Apparoach, Second edition, Tata McGraw-Hill, 2006. (Chapter 1,2,3,4,5,6,7 and 8) 2. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 7 th edition, Wiley-India, 2006. (Chapter 6,7 and 21) |
| **Reference Book:**   1. Harvey M Deital: Operating systems, 3rd Edition, Addison Wesley 2. William Stallings:Operating Systems, 6th Edition, Addison Wesley |

|  |  |  |
| --- | --- | --- |
| **UCS454C** | **MICROCONTROLLERS** | **3-CREDITS** |
| **Hrs/Week :03** |  | **CIE Marks:50** |
| **Total Hrs: 40** |  | **SEE Marks:50** |

|  |  |
| --- | --- |
|  | **Course Outcomes**  At the end of the course the student will be able to , |
| CO 1. | List different addressing modes and instructions of different types. |
| CO 2. | Explain the functionalities of serial communication, interrupts and I/O Interfacing. |
| CO 3. | Develop assembly program and 8051 C program for simple microcontroller based application. |
| CO 4. | Calculate delays generated by delay routines and timers. |
| CO 5. | Design address decoder to interface external memory. |
| CO 6. | Select appropriate mode of operation for programming interface circuits. |
| **Unit I ( 12 Hours)**  **The 8051 Microcontrollers:** Microcontrollers and Embedded systems, Overview of the 8051 family.Pin description of 8051.  **8051 Assembly Language Programming:** Inside the 8051, Introduction to 8051 Assembly Programming, Assembling and running an 8051 program, the program counter and ROM space in the 8051, 8051 data types and directives, 8051 flag bits and PSW register, 8051 register banks and stack.  **Jump, Loop and Call Instructions:** Loop and Jump instructions, Call instructions, Time delay for various 8051 chips. | | |
| **Unit II ( 12 Hours)**  **I/O Port Programming:** 8051 I/O programming, I/O bit manipulation programming.  **8051 Addressing Modes:** Immediate and register addressing modes, Accessing memory using various addressing modes, Bit addresses for I/O and RAM, Extra 128-byte-on-chip RAM in 8052.  **Arithmetic, Logic Instructions and Programs:** Arithmetic Instructions, Signed number concepts arithmetic operations, Rotate instruction and data serialization, BCD, ASCII, and other application programs.  **8051 Programming in C:** Data types and time delay in 8051 C, I/O programming in 8051 C, Logic operations in 8051 C, Data serialization using 8051C. | | |
| **Unit III ( Hours)**  **8051 Timer Programming in Assembly and C:** Programming 8051 timers, counter programming.  **8051 Serial Port Programming in Assembly and C:** Basics of serial communication, 8051 serial port programming in Assembly, Serial port programming in C.  **Interrupts Programming in Assembly and C:** 8051 interrupts, Programming timer interrupts, Programming external hardware interrupts, Programming the serial communication interrupt, Interrupt priority in the 8051/52, Interrupt programming in C. | | |
| **Unit IV ( Hours)**  **8051 Interfacing to External Memory:** Semiconductor Memory, Memory address decoding, 8031/51 interfacing with external ROM, 8051 data memory space, accessing external data memory in 8051 using C.  **8051 interfacing with the 8255:** Programming the 8255, 8255 interfacing, 8051 C Programming for the 8255. Stepper motor interfacing, DC motor interfacing and PWM. | | |
| **Text Books:**   1. Muhammad Ali Mazidi, Janice Gillipse Mazidi and Rolin D. Mckinlay, “ The 8051 Microcontroller and Embedded Systems” using Assembly and C. Pearson 2nd Edition, 2011. | | |
| **Reference Book:**   1. Kenneth J. Ayala, "The 8051 Microcontroller Architecture, Programming and Applications", Penram International, 1996 / Thomson Learning 2005, 2nd Edition. 2. Dr. Uma Rao and Dr. Andhe Pallavi, "The8051 Microcontroller – Architecture, Programming And Applications", Pearson - Sanguine Publishers, Bengaluru, 2009. 3. V Udayshankar, M S Mallikarjunswamy, “8051 Microcontroller: Hardware, Software and Applications”, McGrawHill, New Delhi. | | |

|  |  |  |
| --- | --- | --- |
| **UCS455L** | **Data Structures Using C Lab** | **1-CREDIT** |
| **Hrs/Week :2** |  | **CIE Marks:50** |
|  |  | **SEE Marks:50** |

|  |  |
| --- | --- |
|  | **Course Outcomes**  At the end of the course the student should be able to , |
| CO 1. | Implement various searching and sorting techniques |
| CO 2. | Analyze, design & develop solutions using appropriate data structures for the problem given |

|  |
| --- |
| 1. Develop linked stack ADT and create stack of integer using the ADT’s defined 2. Develop array stack ADT and create stack of students using the ADT’s defined 3. Develop linked Queue ADT and create Queue of floats using the ADT’s defined 4. Develop array Queue ADT and create Queue of strings using the ADT’s defined 5. Create Linked list ADT and use the same to create list of students information 6. Apply following searching algorithm to search the key element 7. From the list of strings using sequential search 8. From the list of integers using binary search 9. From the list of integers using Hashed search 10. Apply following sorting algorithm to sort the list of integers in ascending or descending order based on user choice using 11. Selection 12. Insertion, 13. exchange 14. quick sorts 15. Create binary tree and allow following operations on tree   i)Search an element  ii)Insert an element  iii)Tree is balanced or not  iv)No of occurrences of key element  v)No of nodes, no of leaf nodes, no of intermediate node  vi)Find parent of key node  vii)Traverse in preorder, postorder,inorder,breadth first order  viii)To copy tree   1. Create binary search tree of integers and allow following operations on tree   i)Search an element  ii)Insert an element  iii)Tree is balanced or not  iv)No of occurrences of key element  v)No of nodes, no of leaf nodes, no of intermediate node  vi)Find parent of key node  vii)Traverse in preorder, postorder, inorder, breadth first order  viii)To copy tree  ix ) To print elements in descending order |

|  |  |  |
| --- | --- | --- |
| **UCS456L** | **MICROCONTROLLERS LAB** | **1-CREDIT** |
| **Hrs/Week :02** |  | **CIE Marks:50** |
| **Total Hrs: 30** |  | **SEE Marks:50** |

|  |  |
| --- | --- |
|  | **Course Outcomes**  At the end of the course the student will be able to , |
| CO 1. | Write Assembly and C Programs for 8051 Microcontroller. |
| CO 2. | Develop Applications Using Kiel Compiler and Simulator. |
| CO 3. | Interface I/O Devices using Programmable peripheral interface. |
| CO 4. | Program 8255 in appropriate mode and interface devices. |
| CO 5. | Develop programs for simple microcontroller based applications. |
| **Software Part ( 20 Hours)**   1. Write an ALP to find the frequency of occurrence of a key element in a list of N numbers. The key element is stored in memory location 30H.The total number of elements (N) in a list is stored in a memory location 31H and the numbers are stored in consecutive memory locations from 30 to H. Store the results in register in r7. 2. Write an ALP to find smallest of ‘n’ numbers. 3. Write an ALP to implement decimal up/down counter. Display the count value on port P0.The mode (up/down) of the counter is set at port P1.0 (P1.0=1 up , P1.0=0 down). 4. Write an ALP to read the 10 numbers from the port P1 and store them in to RAM from location 30h.Availability of each numbers is indicated by setting the bit P20.Count the number of elements which are greater than 0fh and display the count values on port P0. 5. Write an ALP to transfer the string stored in external RAM from the location 300h to internal RAM location 30h. 6. Write an ALP to sort number in ascending order. 7. Write a 8051 C program to generate triangular wave. 8. Write an 8051 C program to generate full staircase wave with n number of steps. 9. Write an ALP to compare the list of numbers where one list is available in ROM that is in port 1 space and another is available in RAM. If they are equal send ee to port 0 otherwise 00 to port 0. 10. Write an ALP to count number of zero’s in a 8-bit number used from port-0.If number has even number of zero’s send it to port-2 otherwise to port-1. | | |

|  |
| --- |
| **Hardware Part ( 10 Hours)**   1. Write an assembly language program to implement single digit decimal counter using 7-segment display U16 seven-segment table 2. Write an assembly program to raise the interrupt on INT0 pin whenever INT0 interrupt is raised LED24 must glow (LED24 is connected with p1.5, recognize the interrupt on falling edge) 3. Write an ALP to rotate the stepper motor by 180 in clockwise direction 4. Write an ALP to test dc motor by varying off-time and on-time with different values user can observe different speed on dc motor typically the off values are 30,24,18 & 12 on values are 10,16,12 & 28 5. Write an ALP to test Traffic Light Simulator. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UCS457L** | | | **Database Management Systems Lab** | | **1-CREDITS** |
| **Hours/Week: 03** | | | | **CIE MARKS: 50** | |
| **Exam Hours: 03** | | | | **SEE Marks: 50** | |
|  | | **Course Outcomes**  At the end of the course the student should be able to , | | | |
| CO 1. | | Create and maintain database using SQL. | | | |
| CO 2. | | Query the given database to generate reports. | | | |
| CO 3. | | Design and develop real time database applications. | | | |
| **PART – A**  Design the Database for any one of the following Applications and implement the SQL Queries on designed database.   * 1. Banking System,   2. Employee Organization   3. Inventory Processing System   4. Library Management | | | | |
| 1. Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) using CREATE, ALTER, DROP, INSERT statements 2. Implementing the queries for Insertion, Updation, Deletion operations. Use ROLL BACK, COMMIT & SAVE POINTS Concepts with INSERT, UPDATE, DELETE statements. 3. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSECT, Constraints. 4. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING 5. Creation and dropping of Views. | | | | |
| 1. Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES. 2. Program development using creation of stored functions, invoke functions in SQL Statements. 3. Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables. 4. Develop Programs using BEFORE and AFTER Triggers.   **PART – B**  Develop **Mini Project** on any application. | | | | |

|  |  |  |
| --- | --- | --- |
|  | | |
| **UHS001N** | **FUNDAMENTALS OF QUANTITATIVE APTITUDE**  **AND SOFT SKILLS** | **1-CREDIT** |
| **Hrs/Week :02** |  | **CIE Marks:50** |
| **Total Hrs:13** |  | **SEE Marks:50** |

|  |  |  |
| --- | --- | --- |
|  | | **Course Outcomes**  At the end of the course the student should be able to , |
| CO 1. | | Students are able to explain the range of multiplicative strategies when operating on whole numbers. |
| CO 2. | | Students are able to divide large numbers, by using the rules of divisibility |
| CO3 | | Students are able to get the common multiples and common factors of three numbers |
| CO4 | | Find out the importance of critical thinking; identify the core skills associated with critical thinking. |
| CO5 | | Students are able to learn the importance of the skills for successful businesses and for building relationships by influencing interactions either positively or negatively. |
| CO6 | | Guide students in making appropriate and responsible decisions, to create a desire to fulfil individual goals and to educate students about unproductive thinking, self-defeating emotional impulses, and self-defeating behaviours |
|  | |  |
| **Unit I : Quantitative Aptitude 3 Hours**  Factors and Multiples – Unique Factors, Prime factors, Even factors, Odd factors, Sum of Factors and Product of Factors, Divisibility Rules – Divide the numbers using shortcuts and LCM & HCF – Prime factorization, Division method | |
| **Unit II : Verbal Ability 3 Hours**  Sentence Completion – Read the Sentence, hints, pluses and minuses, structure words, visualize, Para Jumbles – noun-pronoun relationship approach, Acronym approach, Impromptu Speaking – On your own, Speak in a group and Fill in the blanks (Grammar) – Delivery style, Question wording, Question blanks. | |
| **Unit III : Logical Reasoning 3 Hours**  Puzzles – Analytical Puzzles, Math Puzzles, Venn Diagrams – Basics of Set Theory, Operations of Sets and Problem Solving using Venn Diagrams, Binary Logic – Boolean Logics, Logic Gates. | |
| **Unit IV : Soft Skills 4 Hours**  Goal Setting – Types of Settings, SMART Goals, Failure of Goal Settings, Communication Skills – Process of Communication, Levels of Communication, Listening Skills – Steps of Listening, Importance of Listening and why people don’t improve listening skills , SWOT Analysis – Strength, Weakness, Opportunity and Threat, Team Work – benefits of team work, effective team work, basic team dynamics and process of team work. | |
| **Text Book:**  R.S.Aggarwal and Vikas Aggarwal, “ Objective General English”, S.Chand & Company Ltd.  R.S.Aggarwal “ Quantitative Aptitude”, S.Chand & Company Ltd.  R.S.Aggarwal “ Verbal and Non Verbal Reasoning ”, S.Chand & Company Ltd.  Shalini Aggarwal, “Essential Communication Skills”, S.Chand & Company Ltd. | |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Basaveshwar Engineering College (Autonomous), Bagalkot**  **Department of Computer Science and Engineering (CSE)**  **Draft Scheme of Syllabus for B. E. (CSE) programme for 175 credits**  Revised Scheme and Syllabus of teaching **(2018-19 Onwards Admitted Batches)**  **(Effective from the academic year 2020-21)** | | | | | | | | | | |
| **Programme: B.E. COMPUTER SCIENCE AND ENGINEERING** | | | | | | | | | | |
| **V SEMESTER** | | | | | | | | | | |
| **Sl. No** | **Course and**  **Course code** | Course Title | **Teaching Hours /Week** | | | **Examination** | | | |  |
| **Theory Lecture** | **Tutorial** | **Practical** | **Credits** | **CIE Marks** | **SEE Marks** | **Total Marks** |
| **L** | **T** | **P** |
| 1 | UCS551C | Analysis & Design of Algorithms | 3 | 2 | 0 | 4 | 50 | 50 | 100 |  |
| 2 | UCS552C | Finite Automata & Formal Languages | 2 | 2 | 0 | 3 | 50 | 50 | 100 |  |
| 3 | UCS553C | Data Communications | 3 | 2 | 0 | 4 | 50 | 50 | 100 |  |
| 4 | UCS554C | System Software | 2 | 2 | 0 | 3 | 50 | 50 | 100 |  |
| 5 | UCS041E | Artificial Intelligence and Expert Systems | 3 | 0 | 0 | 3 | 50 | 50 | 100 |  |
| 6 | UCS065E | Python Application Programming | 3 | 0 | 0 | 3 | 50 | 50 | 100 |  |
| 7 | UCS555L | Operating Systems Lab | 0 | 0 | 2 | 1 | 50 | 50 | 100 |  |
| 8 | UCS556L | System Software Lab | 0 | 0 | 2 | 1 | 50 | 50 | 100 |  |
| 9 | UHS002N | Advanced Quantitative Aptitude and Soft Skills | 0 | 2 | 0 | 1 | 50 | 50 | 100 |  |
| **TOTAL** | | | **16** | **10** | **04** | **23** | **450** | **450** | **900** |  |
| **AICTE activity Points:** In case students fail to earn the prescribed activity Points, Eighth semester Grade Card shall be issued only after earning the required activity Points. Students shall be admitted for the award of degree only after the release of the Eighth semester Grade Card. | | | | | | | | | | |
|  | | | | | | | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **B.E (COMPUTER SCIENCE AND ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**  **SEMESTER – V** | | | | |
| **ANALYSIS AND DESIGN OF ALGORITHMS** | | | | |
| Course Code: | | UCS551C | CIE Marks | 50 |
| TeachingHours/Week (L:T:P) | | (3:2:0) | SEE Marks | 50 |
| Credits | | 04 | Hours | 48 |
| **Course objectives: Analyze the asymptotic performance of algorithms.**   * Have insight into the basics of various algorithmic design techniques. * To develop proficiency in algorithmic approaches of Brute Force, Divide and Conquer, Decrease and conquer, Greedy and Dynamic programming. | | | | |
| **UNIT -I (12 hours)** | | | | |
| **Introduction:** Notion of Algorithm, Fundamentals of Algorithmic Problem Solving, Important Problem Types, Fundamental Data Structures.  **Fundamentals of the Analysis of Algorithm Efficiency:** Analysis Framework, Asymptotic Notations and Basic Efficiency Classes, Mathematical Analysis of Non-recursive and Recursive Algorithms, Example – Fibonacci Numbers.  **Brute Force:** Selection Sort and Bubble Sort, Sequential Search and Brute-Force String Matching, Exhaustive Search. | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | Remembering,L3 –Applying L4-Analysis | | | |
| **UNIT- II (12 hours)** | | | | |
| **Divide and Conquer:**Mergesort, Quicksort, Binary Search, Binary tree traversals and related properties, Multiplication of large integers and Stressen’s Matrix Multiplication.  **Decrease and Conquer: Insertion Sort, Depth First Search, Breadth First Search, Topological Sorting, Algorithms for Generating Combinatorial Objects.** | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | Remembering,L3 –Applying L4-Analysis | | | |
| **UNIT- III (12 hours)** | | | | |
| **Transform and Conquer:** Presorting, Balanced Search Trees, Heaps and Heapsort, Problem Reduction **Space and Time Tradeoffs:** Sorting by Counting, Input Enhancement in String Matching , Hashing, B-Trees **Dynamic Programming:** Computing a Binomial Coefficient, Warshall’s and Floyd’s Algorithms, Optimal Binary Search Trees. The Knapsack Problem and Memory Functions. | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | Remembering,L3 –Applying L4-Analysis | | | |
| **UNIT- IV (12 hours)** | | | | |
| **Greedy Technique:** Prim’s Algorithm,Kruskal’s Algorithm, Dijkstra’s Algorithm, Huffman Trees.  **Limitations of Algorithm Power:** Lower-Bound Arguments, Decision Trees, Problems  Coping with the Limitations of Algorithm Power: Backtracking, Branch-and-Bound, | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | Remembering,L3 –Applying L4-Analysis | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Course outcomes:**  At the end of the course the student will be able to:   * CO1: Analyze the asymptotic performance of algorithms. * CO2: Demonstrate a familiarity with major algorithms and data structures. * CO3: Implement the algorithms to ascertain their working. * CO4: Apply important algorithmic design paradigms and methods of analysis. * CO5: Synthesize efficient algorithms in common engineering design situations. | | | | |
|  | | | | |
| **Sl No** | **Title of the Book** | **Name of the Author/s** | **Name of the Publisher** | **Edition and Year** |
| **Textbooks** | | | | |
| 1 | Introduction to The Design & Analysis of Algorithms | AnanyLevitin | Pearson Education. | 3rd Edition, 2017 |
| **Reference Books** | | | | |
| 1 | Introduction to Algorithms | Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein | PHI | 2nd Edition, |
| 2 | Computer Algorithms | Horowitz E., Sahni S., Rajasekaran S., | Galgotia Publications | 2001 |
|  | | | | |
| **Web links and Video Lectures:**  1. <https://nptel.ac.in/courses/106/106/106106131/>  2. <https://www.classcentral.com/course/swayam-design-and-analysis-of-algorithms-3984>  3. VTU EDUSAT PROGRAMME – 20 | | | | |
|  | | | | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **B.E (COMPUTER SCIENCE AND ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**  **SEMESTER – V** | | | | | | | | | |
| **FINITE AUTOMATA AND FORMAL LANGUAGES** | | | | | | | | | |
| Course Code | | | | UCS552C | | | CIE Marks | | 50 |
| Teaching Hours/Week (L:T:P) | | | | 26 Teaching + 26 Tutorial (2:2:0) | | | SEE Marks | | 100 |
| Credits | | | | 03 | | | Hours | | 40 |
| **Course objectives:**   * To have an insight into the basic principles of computation including automata, grammars and Turing machines * To develop the proficiency in theoretical foundations of Computer Science. | | | | | | | | | |
| **UNIT -I(6 hours teaching + 6 hours tutorials)** | | | | | | | | | |
| **Introduction To Theory of Computation:** Three basic concepts; some applications. (2 Hours)  **Finite Automata:** Deterministic Finite Accepters; Nondeterministic Finite Accepters; Equivalence of deterministic and Nondeterministic Finite Accepters; Reduction of the number of states in Finite Automata. (4 Hours) | | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | | L1: Remembering, L2: Understanding, L3: Applying | | | | | | |
| **UNIT- II (7 hours teaching + 7 hours tutorials)** | | | | | | | | | |
| **Regular Languages and Regular Grammars**: Regular expressions; Connection between Regular Expression and Regular Languages; Regular Grammars. (3 Hours)  **Properties of Regular Languages:** Closure Properties of Regular Languages; Elementary Questions about Regular Languages; Identifying Nonregular Languages. (4 Hours) | | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | | L1: Remembering, L2: Understanding, L3: Applying, L4: Analyzing | | | | | | |
| **UNIT- III (6 hours teaching + 6 hours tutorials)** | | | | | | | | | |
| **Context-Free Languages:** Context-Free Grammars; Parsing and Ambiguity; (3 Hours)  **Simplification of Context-Free Grammars and Normal Forms:** Methods of Transforming Grammars; Two Important Normal Forms. (3 Hours) | | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | | L1: Remembering, L2: Understanding, L3: Applying, L4: Analyzing | | | | | | |
| **UNIT- IV (7 hours teaching +7 hours tutorials)** | | | | | | | | | |
| **Pushdown Automata:** Nondeterministic Pushdown Automata; Pushdown Automata and Context-Free Languages; Deterministic Pushdown Automata and Deterministic Context-Free Languages. (4 Hours)  **Turing Machines**: The Standard Turing Machine Turing Machine with More Complex Storage: Multitape and Multidimensional Turing Machines (3 Hours) | | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | | L1: Remembering, L2: Understanding, L3: Applying, L4: Analyzing | | | | | | |
| **Course outcomes:**  At the end of the course the student will be able to:   1. Demonstrate a fundamental knowledge of the core concepts in automata theory and formal languages. 2. Prove the properties of languages, grammars and automata with formal mathematical methods; 3. Analyze the closure properties of regular and context-free languages. 4. Design finite automata, pushdown automata, Turing machines for solving language pattern recognition problems. 5. Apply mathematical and formal techniques for solving problems in Computer Science. | | | | | | | | |
|  | | | | | | | | |
| **Sl No** | **Title of the Book** | | | **Name of the Author/s** | **Name of the Publisher** | | **Edition and Year** | |
| **Textbooks** | | | | | | | | |
| 1 | Introduction to Formal Languages and Automata | | | Peter Linz | Jones and Bartlett Student Edition | | 6th Edition, 2016 | |
| **Reference Books** | | | | | | | | |
| 1 | Introduction to Automata Theory, Languages, and Computation, | | | Hopcroft, Motwani, and Ullman | Pearson Education India | | 3rd Edition, 2014 | |
| 2 | Introduction to the Theory of Computation | | | Michael Sipser | Cengage Learning | | 3rd  Edition, 2012 | |
| 3 | Theory of Computer Sciences | | | Korral | McGraw-Hill | | 11th Edition, 2010 | |
| 4 | Automata, Computability and Complexity: Theory and Applications | | | E Rich | Pearson Education India | | 1st Edition, 2012 | |
| 5 | Introduction to languages and the theory of computation. | | | Martin, John C | McGraw-Hill | | 4th Edition, 2013 | |
| 6 | Theory of Computer Science | | | K L P Mishra, N Chandrasekaran | PHI Learning Pvt. Ltd. | | 3rd Edition, 2012 | |
| 7 | Elements of the Theory of Computation | | | H. R. Lewis, C.H. Papadimitriou | Pearson Education, Asia | | 2nd Edition, 2001 | |
| **Web links and Video Lectures:**  1. https://nptel.ac.in/courses/106/105/106105196/  2. https://nptel.ac.in/courses/106/104/106104028/  3.http://nptel.vtu.ac.in/econtent/courses/CSE/CS44/index.php | | | | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **B.E (COMPUTER SCIENCE AND ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**  **SEMESTER – V** | | | | | |
| **Data Communications** | | | | | |
| Course Code | | | UCS553C | CIE Marks | 50 |
| Teaching Hours/Week (L:T:P) | | | (3:2:0) | SEE Marks | 50 |
| Credits | | | 04 | Hours | 52 |
| **Course objectives:**  1. Have insight into the fundamental concepts of Data Communication.  2. Develop proficiencyin Computer Networking concepts. | | | | | |
| **UNIT-I(13 hours)** | | | | | |
| **Introduction:** Data Communications; Networks; the Internet; Protocols and Standards.**Network Models:** Layered tasks; The OSI Model, Layers in the OSI model; TCP/IPProtocol Suite, Addressing.**Data and Signals:**Analog and digital signals; Periodic Analog Signals, Digital Signals, Transmission impairment; Data rate limits; Performance. | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | L1:Remembering, L2:Understanding, L3,:Applying, L4: Analyzing | | | | |
| **UNIT- II (13 hours)** | | | | | |
| **Digital Transmission, Analog Transmission and Multiplexing:** Digital-to-Digital conversion; Analog-to-Digital conversion: PCM; Transmission modes, Digital - to - Analog conversion; Analog - to - Analog conversion; Multiplexing.**Transmission Media:** Guided media, unguided media: Wireless. | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | L1:Remembering, L2:Understanding, L3,:Applying, L4: Analyzing | | | | |
| **UNIT- III (13 hours)** | | | | | |
| **Error Detection and Correction:**Introduction to Error Detection and Correction; Block Coding; Linear Block Codes; Cyclic codes, Checksum.**Data Link Control:** Framing; Flow and Error control; Protocols; Noiseless channels; Noisy channels; HDLC; Point-to-point Protocol. | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | L1:Remembering, L2:Understanding, L3,:Applying, L4: Analyzing | | | |
| **UNIT- IV (13 hours)** | | | | | |
| **Multiple Accesses:** Random Access; Controlled Access; Channelization.Ethernet: EEE standards; Standard Ethernet and changes in the standard; Fast Ethernet; Gigabit Ethernet.**Wireless LANs and Connection of LANs:** IEEE 802.11; Bluetooth. Connecting devices; Backbone Networks, Virtual LANs. | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | L1:Remembering, L2:Understanding, L3,:Applying, L4: Analyzing | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Course Outcomes:**  At the end of the course the student will be able to:  CO1: Identifying various design parameters, and their influence on node/link utilization and performance.  CO2: Explain the concept of Data Communication and networks, layered architecture and their applications.  CO3: Apply the concepts of Digital Transmission, Analog Transmission and Multiplexing.  CO4: Analyze MAC layer protocols and LAN technologies  CO5: Evaluate data communication link considering elementary concepts of data link layer protocols for error detection and correction. | | | | |
| **Sl No** | **Title of the Book** | **Name of the Author/s** | **Name of the Publisher** | **Edition and Year** |
| **Textbooks** | | | | |
| 1 | Data Communications and Networking. | Behrouz A. Forouzan | Tata McGraw-Hill | 4th Edition, 2017 |
| **Reference Books** | | | | |
| 1 | Communication Networks - Fundamental Concepts and Key architectures. | Alberto Leon-Garcia and Indra Widjaja | Tata McGraw-Hill | 2nd Edition,2004 |
| 2 | Data and Computer Communication. | William Stallings | Pearson Education | 8th Edition,2007 |
| 3 | Computer Networks A Systems. Approach | Larry L. Peterson and Bruce S. David | Elsevier | 4th Edition,2007 |
| 4 | Computer and Communication Networks | Nader F. Mir | Pearson Education | 2nd Edition,2007 |
|  | | | | |
| **Web links and Video Lectures:**  1. https://nptel.ac.in/courses/106/105/106105082/  2. http://www.nptelvideos.in/2012/11/data-communication.html  3. <http://www.nptelvideos.com/course.php?id=399> | | | | |
|  | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **B.E (COMPUTER SCIENCE AND ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**  **SEMESTER – V** | | | | | |
| **SYSTEM SOFTWARE** | | | | | |
| **Course Code** | | | UCS554C | **CIE Marks** | 50 |
| **TeachingHours/Week (L:T:P)** | | | (2:2:0) | **SEE Marks** | 50 |
| **Credits** | | | 03 | **Hours** | 40 |
| **Course Learning objectives(CLO):**  At the end of the course student will learn/understand/ think/experience/appreciate:   1. To have insight into types of system softwares, machine architectures, lex and yacc programming 2. To develop the proficiency in Design of assemblers, compilers loaders and linkers , macro processors | | | | | |
| **UNIT - I (10 hours)** | | | | | |
| **Machine Architecture:**  Introduction, System Software and Machine Architecture, Simplified Instructional Computer (SIC) - SIC Machine Architecture, SIC/XE Machine Architecture, SIC Programming Examples, Traditional (CISC) Machines - VAX Architecture, RISC Machines - Ultra SP ARC Architecture.  **Assemblers**:  Basic Assembler Function - A Simple SIC Assembler, Assembler Algorithm and Data Structures, Machine Dependent Assembler Features - Instruction Formats & Addressing Modes, Program Relocation, Machine Independent Assembler Features - Literals, Symbol-Definition Statements, Expression, Program Blocks, Control Sections and Programming Linking. | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | Remembering, | | | |
| **UNIT- II (10 hours)** | | | | | |
| **Loaders And Linkers:**  Basic Loader Functions - Design of an Absolute loader, A Simple Bootstrap Loader, Machine-Dependent Loader Features - Relocation, Program Linking, Algorithm and Data Structures for a Linking Loader, Machine-Independent Loader Features - Automatic Library Search, Loader Options, Loader Design Options - Linkage Editor, Dynamic Linkage, Bootstrap Loaders. | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | Remembering, | | | |
| **UNIT- III (10 hours)**  **Macro Processor**:  Basic Macro Processor Functions - Macro Definitions and Expansion, Macro Processor Algorithm and Data Structures, Machine-Independent Macro Processor Features - Concatenation of Macro Parameters, Generation of Unique Labels, Conditional Macro Expansion, Keyword Macro Parameters, Macro Processor Design Options - Recursive Macro Expansion | | | | | |
| **Lex And Yacc**: The Simplest Lex Program, Recognizing Words with LEX, Grammars, Parser-Lexer Communication, A YACC Parser, The Rules Section, Running LEX and YACC, LEX and Hand- Written Lexers, Using LEX - Regular Expression, Examples of Regular Expressions, A Word Counting Program, Using YACC - Grammars, Recursive Rules, Shift/Reduce Parsing, What YACC Cannot Parse, A YACC Parser - The Definition Section, The Rules Section, Symbol Values and Actions, The LEXER, Compiling and Running a Simple Parser, Arithmetic Expressions and Ambiguity, Variables and Typed Tokens. | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | Remembering, | | | | |
| **UNIT- IV (10 hours)** | | | | | |
| **Compilers**:  Basic Compiler Function - Grammars, Lexical Analysis, Syntactic Analysis, Code Generation, Machine Dependent Compiler Features ­Intermediate Form of the Program. Machine-Dependent Code Optimization, Machine Independent Compiler Features - Structured Variables, Machine Independent Code Optimization, Storage Allocation, Block Structured Languages. | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | Remembering, | | | | |
|  | | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Course outcomes:**  At the end of the course the student will be able to:  **CO1**: List and define features/concepts of machine architectures and system softwares.  **CO2**: Explain characteristics/concepts/basic operations of machines architectures, system softwares and Lex and Yacc tools.  **CO3**: Write programs to implement simple assembler, loader, linker, macroprocessor, lexical analyzer and syntactic analyzer.  **CO4**: Compare and contrast types of software, machine architectures, system software and Lexical and syntactic analyzer.  **CO5**: Analyze, Design and implement system software for different architectures | | | | |
|  | | | | |
| **Sl No** | **Title of the Book** | **Name of the Author/s** | **Name of the Publisher** | **Edition and Year** |
| **Textbooks** | | | | |
| 1 | System Software – An Introduction to Systems Programming | Leyland.L.Beck | Pearson Education | 3rdEdition, 2012 |
| 2 | Lex and Yacc | John.R.Levine , Tony Mason and Doug Brown | O’Reilly, SPD. | 1999 |
| **Reference Books** | | | | |
| 1 | System Programming and Operating Systems | D.M.Dhamdhere | TMH | 2nd Edition, 1999 |
|  | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **B.E (COMPUTER SCIENCE AND ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**  **SEMESTER – V** | | | |
| **Operating Systems Lab** | | | |
| **Course Code** | **UCS555L** | **CIE Marks** | **50** |
| **TeachingHours/Week (L:T:P)** | **2** | **SEE Marks** | **50** |
| **Credits** | **1** | **Hours** | **50** |

**Course objectives:**

* To have insights into design and implementation of resource management policies of operating systems.
* To have proficiency in concurrent programming.

|  |  |  |
| --- | --- | --- |
| Week Number | Work to be done |  |
| 1 | 1. Implementation of scheduling policies |  |
| 2 | 1. Implementation of memory allocation techniques. |  |
| 3 | Developing solutions for deadlock problems. |  |
| 4 | Implementation of page replacement policies. |  |
| 5 | Developing concurrent applications using processes(Petersons algorithm). |  |
| 6 | Demonstration of synchronization using semaphores. |  |
| 7 | Implementation of Unix like shell commands. |  |
| 8 | Developing concurrent applications using Threads. |  |

**Course outcomes:**

At the end of the course the student should be able

* Simulate and demonstrate different functionalities of operating system
* Implement Unix like Shell commands.
* Develop simple applications using concurrent programming.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SYSTEMS SOFTWARE LABORATORY** | | | | | |
| **Sub Code** | **:** | **UCS556L** | **Credits** | **:** | **1** |
| **Hours/Week** | **:** | **02** | **CIE MARKS** | **:** | **50** |
| **Exam Hours** | **:** | **03** | **SEE Marks** | **:** | **50** |

1. **Course learning Objectives:**

At the end of the course student will learn/practice/think/experience/appreciate:

1. To have insight into Design and implement of system software using C or C++.
2. To have proficiency in Design and implementation of scanners using Lex tool

And implementation of parser using Yacc tool.

1. **Course outcomes:**

At the end of the semester student should be able to:

1. Implement the system software such as assembler, loader and linker etc using C or C++.
2. Design and write Lex program to implement lexical analyzer for given problem statement

And Yacc program to implement parser for the given structure recognition

1. Modify the existing design of the algorithm of system softwares to enhance the efficiency.

**Part I**

1. Write a C program to implement pass one of two pass assembler.
2. Write a C program to implement absolute loader.
3. Write a C program to implement pass one of two pass macroprocessor.
4. Write a C program to implement pass two of two pass macroprocessor
5. Write a C program to generate a lexical analyzer to identify the C keywords
6. Design recursive descent parsers for parsing pascal read and write statements.

**Part II**

Design lexical analyzers using Lex tool to accomplish the following.

1. Design Lexical analyzer to count the no of occurrences of the words from a given text file.The program should accept the text file and list of words as input.
2. Design Lexical analyzer to count no of positive numbers and negative numbers from the input given.
3. Design Lexical analyzer to count number of printf and scanf statements and replace them by sprintf and sscanf respectively.
4. Design Lexical analyzer to count number of integers , float, double, char variable from C declaration statements
5. Design Lexical analyzer to count number of blank spaces lines, characters, words from a given text file.
6. Design Lexical analyzer to check whether a given simple arithmetic operation is valid or not. If valid print number of positive, negative, multiplication and division operators separately

**Part III**

Design parsers using Yacc tool to accomplish the following.

1. Design parser using Yacc tool to test the validity of a simple expression involving operators ‘+’,’-‘,’/’,’\*’.
2. Design parser using Yacc tool to evaluate the given arithmetic expression involving operators ‘+’,’-‘,’/’,’\*’.
3. Design parser using Yacc tool to recognize a valid variable which starts with a letter followed by any number of letters and digits.The length of the identifier should not exceed 15.
4. Design parser using Yacc tool to recognize the grammar an b where n>=10.
5. Design parser using Yacc tool to recognize the validity of nested if statements and also display the number of levels of nesting

**Note:**

**Continuous Internal Evaluation (50 marks):**

Marks are based on execution of assignments and lab internal test. The marks are distributed as below;

1. 30 marks for lab assignment execution.
2. 20 marks for lab internal test.

**Semester End Examination (50 marks):**

In semester end examination two questions will be given. One from Part-I and Part-III respectively.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **B.E (COMPUTER SCIENCE AND ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**  **SEMESTER – VI** | | | | |
| **ARTIFICIAL INTELLIGNCE AND EXPERT SYSTEMS** | | | | |
| Course Code | | UCS041E | CIE Marks | 50 |
| TeachingHours/Week (L:T:P) | | 40 Teaching (3:0:0) | SEE Marks | 100 |
| Credits | | 03 | Hours | 40 |
| Course Objectives:   * To obtain a thorough knowledge of various knowledge representation schemes. * To have an overview of various AI applications. * To study various heuristic search algorithms * To know about Expert system tools and applications. | | | | |
| **Unit -1 (10 hours)** | | | | |
| **1. What is AI?** The AI Problems, Underlying assumptions, AI technique, Level of the model, Criteria for success (1.1 to 1.5 from Rich and Knight) 4 Hours  **2. Problems, problem spaces and search** Problem as a state space search, Production systems, Problem characteristics, Production system characteristics, Issues in the design of search problems, additional problems (2.1 to 2.6 from Rich and Knight) 6 Hours | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | L1: Remembering, L2: Understanding | | | |
| **UNIT- II (10 hours)** | | | | |
| **3. Knowledge representation issues** Representation and mappings, approaches to knowledge representation, (4.1 to 4.2 from Rich and Knight), Syntax and semantics for Propositional logic (4.2 from D. W. Patterson) 4 Hours  **4. Using predicate logic** Representing simple facts in logic, representing instance and is-a relationships, computable functions and predicates, resolution, natural deduction (5.1 to 5.5 from Rich and Knight) 6 Hours | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | L1: Remembering, L2: Understanding, L3: Applying, L4: Analyzing | | | |
| **UNIT- III (10 hours)** | | | | |
| **5. Search and control strategies** Introduction, Generate and Test, Hill Climbing, Simulated annealing, (3.1, 3.2 from Rich and Knight), Informed search, Searching And-Or graphs (9.5, 9.6 from D. W. Patterson) 5 Hours  **6. Planning** Overview, an example domain: The Blocks world, Components of a planning system, goal stack planning, non-linear planning using constraint posting,(13.1 to 13.5 from Rich and Knight) 5 Hours | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | L1: Remembering, L2: Understanding, L3: Applying, L4: Analyzing | | | |
| **UNIT- IV (10 hours)** | | | | |
| **7. Expert system architectures** Introduction, Applications of Expert Systems, Roles of expert systems, Rule-based system architectures, Non-production system architectures, Dealing with uncertainty, Knowledge acquisition and validation, Knowledge system building tools, Expert System Shells, Case studies: MYCIN, RI(15.1 to 15.6 from D. W. Patterson) | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | L1: Remembering, L2: Understanding, L3: Applying | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Course outcomes:**  At the end of the course, the students will be able to  1. Identify problems that are amenable to solution by AI methods and identify appropriate  methods to solve a given problem.  2. Illustrate the representation of knowledge and inference for a variety of problems requiring machine intelligence.  3. Analyze various control strategies and solve problems using search techniques  4. Design intelligent systems for simple AI applications.  5. Demonstrate the knowledge of expert systems and intelligent planning. | | | | | |
|  | | | | | |
| **Sl No** | **Title of the Book** | | **Name of the Author/s** | **Name of the Publisher** | **Edition and Year** |
| **Textbooks** | | | | | |
| 1 | Artificial Intelligence | Elaine Rich, Kevin Knight and Shivashankar B. Nair | | TMH Education (P) Ltd., New Delhi | 3rdEdition, 2010 |
| 2 | Artificial Intelligence and Expert Systems | Dan W. Patterson | | PHI Learning (P) Ltd., New Delhi | 2008 |
| **Reference Books** | | | | | |
| 1 | Artificial Intelligence: A modern approach | Stuart Russell and Peter Norvig | | Pearson, | 3rd edition, 2016 |
| 2 | Principles of Artificial Intelligence | Nilson N. J. | | Springer Verlag, | 1980 |
| 3 | Introduction to Artificial Intelligence | Eugene Charniak and Drew McDermot | | Addison-Wesley | 1998 |
| 4 | Introduction To Expert Systems | Peter Jackson | | Pearson Education | 3rd Edition, 2007 |
| 5 | Artificial Intelligence | Deepak Khemani | | Tata Mc Graw Hill Education | 2013 |
| 6 | Artificial Intelligence – Structures and Strategies for Complex Problem Solving | George F. Luger | | Addison-Wesley | 5th Edition, 2005. |
| 7 | Artificial Intelligence application programming | M. Tim Jones | | Dreamtech Press | 2nd Edition,2006 |
| 8 | Introduction to Artificial Intelligence | RajendraAkerkar, | | PHI Learning | 2nd Edition,2014 |
| 9 | Artificial Intelligence | Saroj Kaushik | | Cengage Learning India Pvt Ltd | 2011 |
|  | | | | | |
| **Web links and Video Lectures:**  1. <https://nptel.ac.in/courses/106105077/> (NPTEL course coordinated by IIT Kharagpur)  2. <https://www.mooc-list.com/course/introduction-artificial-intelligence-ai-coursera> (by CourseEra) | | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **B.E (COMPUTER SCIENCE AND ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**  **ELECTIVE** | | | | |
| **Python Application Programming** | | | | |
| **Course Code** | | **UCS065E** | **CIE Marks** | **50** |
| **TeachingHours/Week (L:T:P)** | | **(3:0:0)** | **SEE Marks** | **50** |
| **Credits** | | **03** | **Hours** | **40** |
| **Course objectives:**  1.To acquire programming skills in core Python.  2.To acquire Object Orientation Skills in Python  3.To develop the skill of designing Graphical user Interfaces and networking in Python  4.To develop the ability to write database applications in python | | | | |
| **Unit -1 (10 hours)** | | | | |
| **Datatypes in python:**comments in python, Docstrings, How python sees variables, Datatypes in python, Sequences in python,Literals in python,Determing the data type of a variable, Identifiers and reserved words, Naming conventions in python  **Operators in Python:** Operator,operator precedence and associativity, Mathematical functions  **Input and Output:**Output statements, Input statements, Command Line arguments  **Control Statements**  **Strings and Characters** | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | L1- Remembering,L2-,L3-Apply,L4-Analyze | | | |
| **UNIT- II (10 hours)** | | | | |
| **Functions:**Defining a function,calling a function, Returning Results from a function, Returning multiple values from a function,Formal and actual arguments,local and global variables,passing a group of elements to a function,recursive functions,the special variable \_\_name\_\_  **Lists and tuples: lists,tuple**  **Dictionaries**  **Exceptions:**exceptions,exception handling,types of exceptions,user defined exceptions  **Files in python:**files,types of files in python,opening a file,closing a file,working with text files containing strings,working with binary files,pickle in python | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | L1- Remembering,L2-,L3-Apply,L4-Analyze,L5-Evaluate | | | |
| **UNIT- III (10 hours)** | | | | |
| **Regular Expressions in python**  **Object Orineted Programming:**Classes and Objects, Creating Classes in Python, Creating Objects in Python, The Constructor Method, Classes with Multiple Objects, Class Attributes versus Data Attributes, Encapsulation, Inheritance, The Polymorphism  **Networking in python** | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | L1- Remembering,L2-,L3-Apply,L4-Analyze,L5-Evaluate | | | |
| **UNIT- IV (10 hours)** | | | | |
| **Threads**  **Graphical user Interfaces**  **How to work with Database**:How to use SQLite Manager to work with a database, How to use python to work with database | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | L1- Remembering,L2-,L3-Apply,L4-Analyze,L5-Evaluate,L6-Create | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Course Outcomes:**  At the end of the course, students are able to:   1. Explain syntax and semantics of different statements and functions in Python. 2. Demonstrate the use of strings, files, lists, dictionaries and tuples in simple applications. 3. Write simple applications using regular expressions, multiple threads. 4. Build simple database applications with GUI. 5. Analyze the given problem and select appropriate data types and modules to develop the solution. | | | | | | |
|  | | | | | | |
| **Sl No** | **Title of the Book** | **Name of the Author/s** | | **Name of the Publisher** | | **Edition and Year** |
| **Textbooks** | | | | | | |
| 1 | Core Python Programming | Dr. R.Nageswawa Rao | Dreamtech press | | | 2nd Edition 2018 |
| Chapter Numbers:3,4,5,6,8,9,10,11,16,17,18,21,22,23,24 | | | | | | |
| 2 | Introduction to  Python Programming | Gowrishankar S.  Veena A. | CRC Press  Taylor & Francis Group | | | 1st Edition 2019 |
| Chapter Number: **11** | | | | | | |
| 3 | Python Programming | Michael Urban and Joel Murach | Mike Murach  Elizabeth Drake | | | 1st Edition,2016 |
| Chapter Number: **17** | | | | | | |
| **Reference Books** | | | | | | |
| 1. | Learning Python | B Nagesh Rao Python | | | Cyberplus Publication | 1 edition 17 May 2017 |
| 2 | Core Python Applications Programming | Wesley J. Chun | | | Pearson Education India, | Third Edition, 2015. |
| 3 | Python Programming  for the AbsoluteBeginner | Michael Dawson | | | Delmar Cengage Learning | 3rd edition (1 January 2010) |
| 4 | Python Programming using problem solving approach | Reema Thareja | | | Oxford university press, | 1st Edition 2017 |
| 5 | Python for Everybody: Exploring Data Using Python 3 | Charles R. Severance | | | CreateSpace Independent Publishing Platform | 1st Edition, 2016. |
|  | | | | | | |
| **Web links and Video Lecture:**   1. <http://do1.drchuck.com/pythonlearn/EN_us/pythonlearn.pdf> 2. https://www.python.org/ 3. <https://www.pdfdrive.com/introduction-to-python-programming-d176341873.html> 4. <https://www.pdfdrive.com/python-programming-for-the-absolute-beginner-e34494394.html> 5. <https://edubookpdf.com/programming/murachs-python-programming.html> 6. https://www.youtube.com/watch?v=rfscVS0vtbw 7. <https://www.youtube.com/watch?v=vaysJAMDaZw> 8. <https://www.youtube.com/playlist?list=PLS1QulWo1RIaJECMeUT4LFwJ-ghgoSH6n> 9. https://www.youtube.com/playlist?list=PL6gx4Cwl9DGAcbMi1sH6oAMk4JHw91mC\_ 10. https://www.youtube.com/playlist?list=PLTTTcaxrixZSh3TyvoEoTTbEHyS4c6Su7 | | | | | | |
|  | | | | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **BASAVESHAWAR ENGINEERING COLLEGE [AUTONOMOUS] BAGALKOT**  **Scheme of Teaching and Examination 2018 – 19**  **(Effective from the academic year 2020-21)** | | | | | | | | | | | | | |
| **Programme:** B.E. **COMPUTER SCIENCE AND ENGINEERING** | | | | | | | | | | | | | |
| **VI SEMESTER** | | | | | | | | | | | | | |
| **Sl. No** | **Course and**  **Course code** | Course Title | | **Teaching Hours /Week** | | | | **Examination** | | | | |  |
| **Theory Lecture** | | **Tutorial** | **Practical** | **Credits** | | **CIE Marks** | **SEE Marks** | **Total Marks** |
| **L** | | **T** | **P** |
| 1 | UCS651C | Computer Networks | | 2 | | 2 | 0 | 3 | | 50 | 50 | 100 |  |
| 2 | UCS652C | Computer Graphics & Visualization | | 2 | | 2 | 0 | 3 | | 50 | 50 | 100 |  |
| 3 | UCS653C | Software Engineering | | 2 | | 2 | 0 | 3 | | 50 | 50 | 100 |  |
| 4 |  | Elective II | | 3 | | 0 | 0 | 3 | | 50 | 50 | 100 |  |
| 5 |  | Open Elective II | | 3 | | 0 | 0 | 3 | | 50 | 50 | 100 |  |
| 6 | UCS654H | Management & Entrepreneurship | | 2 | | 2 | 0 | 3 | | 50 | 50 | 100 |  |
| 7 | UCS656L | Computer Graphics Lab | | 0 | | 0 | 2 | 1 | | 50 | 50 | 100 |  |
| 8 | UHS003N | Career planning and Professional Skills | | - | | 2 | - | 1 | | 50 | 50 | 100 |  |
| 9 | UCS657P | Mini-project | | -- | | -- | 3 | 3 | | 50 | 50 | 100 |  |
| 10 | -- | Internship | To be carried out during the vacation/s of VI and VII semesters | | | | | | | | | | |
| **TOTAL** | | | | | **14** | **10** | **5** | | **23** | **450** | **450** | **900** |  |
|  | | | | | | | | | | | | | |
| **Open Elective –II** | | | | | | | | | | | | | |
| **Internship:** All the students admitted to III year of BE/B.Tech shall have to undergo mandatory internship of 6 weeks during the vacation of VI and VII semesters . A University examination shall be conducted during VII semester and the prescribed credit shall be included in VII semester. Internship shall be considered as a head of passing and shall be considered for the award of degree. Those, who do not take-up/complete the internship shall be declared fail and shall have to complete during subsequent examination after satisfying the internship requirements.  **AICTE activity Points:** In case students fail to earn the prescribed activity Points, Eighth semester Grade Card shall be issued only after earning the required activity Points. Students shall be admitted for the award of degree only after the release of the Eighth semester Grade Card. | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **B.E (COMPUTER SCIENCE AND ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**  **SEMESTER – VI** | | | | | | | | | | |
| **COMPUTER NETWORKS** | | | | | | | | | | |
| Course Code | | | UCS651C | | | | | CIE Marks | | 50 |
| Teaching Hours/Week (L:T:P) | | | (2:2:0) | | | | | SEE Marks | | 50 |
| Credits | | | 03 | | | | | Hours | | 48 |
| **Course Objectives**   |  | | --- | | 1. Have insight into the basic taxonomy and terminology of the computer networking area. 2. Develop proficiency in specific areas of networking such as the design andmaintenance of individual networks. | | | | | | | | | | | |
| **UNIT -I (12 hours)** | | | | | | | | | | |
| **Network Layer:** IPv4 Addresses, IPv6 Addresses. Internetworking, Packet format of IPv4 and IPv6, Transition from IPv4 to IPv6. Address Mapping, ICMP, Delivery, Forwarding, Unicast Routing Protocols. | | | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | L1:Remembering, L2:Understanding, L3,:Applying, L4: Analyzing | | | | | | | | |
|  | | | | | | | | | | |
| **UNIT- II (12 hours)** | | | | | | | | | | |
| **Transport Layer:** Process-to-Process Delivery, UDP, TCP, and SCTP. Data traffic, Congestion, Congestion Control, Two Examples, Quality of Service, Techniques to improve QoS, QoS in Switched Networks. | | | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | L1:Remembering, L2:Understanding, L3,:Applying, L4: Analyzing | | | | | | | | |
| **UNIT- III (12 hours)** | | | | | | | | | | |
| **Application Layer:** Name Space, Domain Name Space, Distribution of Name Space, DNS In The Internet, Resolution. DNS Messages. Remote Logging, Electronic Mail, File Transfer. Architecture of WWW, Web Documents, HTTP. | | | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | L1:Remembering, L2:Understanding, L3,:Applying, L4: Analyzing | | | | | | | | |
| **UNIT- IV (12 hours)** | | | | | | | | | | |
| **Network Management and Security:** Network Management System. Digitizing Audio and Video, Audio and Video Compression, Streaming Stored Audio/Video, Streaming Live Audio/Video, Real-Time Interactive Audio/Video, RTP, RTCP, Voice Over IP. Introduction to Cryptography, Symmetric-Key Cryptography, Asymmetric-key Cryptography. | | | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | L1:Remembering, L2:Understanding, L3,:Applying, L4: Analyzing | | | | | | | | |
| **Course Outcomes**  At the end of the course the student will be able to:   |  | | --- | | **CO1:**Explain the fundamental concepts of Computer Networks. | | **CO2:** Analyze different network protocols. | | **CO3: A**pply techniques for efficient handling of Computer Networks. | | **CO4:** Formulate Routing and Congestion Control Algorithms. | | **CO5:** Implement Application Layer and Network Security protocols. | | | | | | | | | | | |
| **Sl No** | | **Title of the Book** | | | | **Name of the Author/s** | **Name of the Publisher** | | | **Edition and Year** | |
| **Textbooks** | | | | | | | | | | | |
| 1 | | Data Communications and Networking. | | | | Behrouz A. Forouzan | | Tata McGraw-Hill | | 4th Edition, 2017 | |
| **Reference Books** | | | | | | | | | | | |
| 1 | | Computer Networking-A top-down approach featuring the Internet | | | James F. Kurose, Keith W. Ross | | | Pearson Education | | 3rd Edition, 2018 | |
| 2 | | Data and Computer Communication | | | William Stallings | | | Pearson Education | | 8th Edition, 2016 | |
| 3 | | Computer Networks A  Systems Approach | | | Larry L. Peterson and Bruce S. David | | | Elsevier | | 4th Edition, 2017 | |
| 4 | | Communication Networks | | | Garcia Leon And Widjaja | | | Tata Mcgraw-Hill,. | | 15th Edition, 2019 | |
|  | | | | | | | | | | | |
| **Web links and Video Lectures:**  1. http://nptel.vtu.ac.in/econtent/CSE.php  2. https://nptel.ac.in/courses/106/105/106105081/  3. https://nptel.ac.in/courses/106/106/106106091/  4. <http://nptel.vtu.ac.in/econtent/courses/CSE/CS64/index.php> | | | | | | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **B.E (COMPUTER SCIENCE AND ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**  **SEMESTER – VI** | | | | | | |
| **COMPUTER GRAPHICS AND VISUALIZATION** | | | | | | |
| Course Code | | | | **UCS652C** | CIE Marks | 50 |
| TeachingHours/Week (L:T:P) | | | | (2:2:0) | SEE Marks | 50 |
| Credits | | | | 03 | Hours | 40 |
| **Course objectives:**   * Have insight into concepts of computer graphics hardware architecture and its applications. * Have proficiency in 2D and 3D geometric transformations, visualization and interactive graphics applications using OpenGL API. | | | | | | |
| **UNIT -I (10 hours)** | | | | | | |
| **Overview of Graphics Systems:** Video Display Devices, Raster-Scan Displays, GraphicsWorkstations and Viewing Systems, Introduction to OpenGL, **Graphics Output Primitives :**Coordinate Reference Frames, Specifying A Two-Dimensional World-Coordinate Reference Frame in OpenGL, OpenGL Point Functions, OpenGL Line Functions, Line drawing algorithms:Bresenham’s Line-Drawing Algorithm, OpenGL Curve Functions, Circle generating Algorithms: Midpoint Circle Algorithm, Fill-Area primitives, OpenGL Polygon Fill-Area Functions, OpenGL Vertex Arrays, Pixel-Array Primitives, OpenGL Pixel-Array Functions, Character Primitives, OpenGL Character Functions, OpenGL Display Lists, OpenGL Display-Window Reshape Function, **Attributes of Graphics Primitives**: OpenGL State Variables, Color and Grayscale, OpenGL Color Functions, OpenGL Point-Attribute Functions, OpenGL Line-Attribute Functions. | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | L1: Remembering L2: Understanding L3: Applying L4: Analysing | | | | |
| **UNIT- II (10 hours)** | | | | | | |
| **Interactive Input Methods and Graphical User Interfaces:** Graphical Input Data, Logical Classification of Input Devices, Input Functions for Graphical Data, Interactive Picture-Construction Techniques, OpenGL Interactive Input-Device Functions , OpenGL Menu Functions, Designing a Graphical User Interface  **Geometric Transformations-1:** Basic Two-Dimensional Geometric Transformations, Matrix Representations and Homogeneous Coordinates, Inverse Transformations, Two-Dimensional Composite Transformations, Other Two-Dimensional Transformations, Raster Methods for Geometric Transformations, OpenGL Raster Transformations, Transformations between Two-Dimensional Coordinate Systems. | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | L1: Remembering L2: Understanding L3: Applying L4: AnalysingL5: Evaluating L6: Creating | | | | | |
| **UNIT- III (10 hours)** | | | | | | |
| **Geometric Transformations-2:**Geometric Transformations in Three-Dimensional Space, Three-Dimensional Translation, Three-Dimensional Rotation, Three-Dimensional Scaling, Composite Three Dimensional Transformations, Other Three Dimensional Transformations, Transformations between Three Dimensional Coordinate Systems, Affine Tranformations, OpenGL Geometric Transformations Functions.  Two-Dimensional Viewing: The Two-Dimensional Viewing Pipeline, The clipping Window, Normalization and Viewport Transformations, OpenGL Two-Dimensional Viewing Functions, Clipping Algorithms, Two-Dimensional Point Clipping, Two-Dimensional Line Clipping: Cohen-Sutherland line Clipping, Polygon Fill-Area Clipping: Sutherlan-Hodgman Polygon Clipping, Curve Clipping, Text Clipping. | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | | L1: Remembering L2: Understanding L3: Applying L4: AnalysingL5: Evaluating L6: Creating | | | |
| **UNIT- IV (10 hours)** | | | | | | |
| **Viewing:** Classical and Computer Viewing, Viewing with a Computer, Positioning of the Camera, Simple Projections, Projections in OpenGL, Hidden-Surface Removal, Interactive Mesh Displays, Parallel-Projection Matrices, Perspective-Projection Matrices, Projections and Shadows.  **Shading:** Light and Matter, Light Sources, The Phong Reflection Model, Computation of Vectors, Polygonal Shading, Approximation of a Sphere by Recursive Subdivision, Light Sources in OpenGL, Specification of Materials in OpenGL, Shading of the Sphere Model, Global Illumination. | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | | L1: Remembering L2: Understanding L3: Applying L4: AnalysingL5: Evaluating L6: Creating | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Course outcomes:**  At the end of the course the student will be able to:   * CO1: Explain fundamental concepts of computer graphics and visualization. * CO2: Implement the graphics algorithms to draw geometric primitives. * CO3: Develop an interactive 2D and 3D graphics applications. * CO4: Illustrate the animations of graphics models using geometric transformation functions. * CO5: Construct the graphical model with lighting and shading patterns. | | | | |
|  | | | | |
| **Sl No** | **Title of the Book** | **Name of the Author/s** | **Name of the Publisher** | **Edition and Year** |
| **Textbooks** | | | | |
| 1 | Computer Graphics with OpenGL | Donald Hearn and Pauline Baker | Pearson Education | 3rd Edition ,2004 |
| 2 | Interactive Computer Graphics A Top-Down Approach using OpenGL | Edward Angel | Addison-Wesley | 5th Edition, 2008 |
| **Reference Books** | | | | |
| 1 | Computer Graphics using OpenGL | F.S.Hill Jr. | Pearson Education | 2nd Edition, 2001 |
| 2 | Computer Graphics | James D. Foley, Andries Van Dam, Steven K Feiner, John F. Hughes | Addison-wesley | 1997 |
| **Web links and Video Lectures:**  1. Manual - Computer Graphics: Programming approach using Open-GL, Dr. S.V. Saboji  2.https://nptel.ac.in/course.html  3. <http://www.cse.iitm.ac.in/~vplab/computer_graphics.html>  4. <https://www.classcentral.com/course/edx-computer-graphics-548> | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **B.E (COMPUTER SCIENCE AND ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**  **SEMESTER – VI** | | | | |
| SOFTWARE ENGINEERING | | | | |
| Course Code | | **UCS653C** | CIE Marks | 50 |
| Teaching Hours /Week (L:T:P) | | (2:2:0) | SEE Marks | 50 |
| Credits | | 03 | Hours | 52 |
| **Course objectives:**   * To have insight in the core principles and practices of software engineering for systematic development of non-trivial software systems. * To have proficiency in the design, development, validatation, testing and managingof the software systems for its overall efficiency. | | | | |
| **UNIT -I (13 hours)** | | | | |
| **OVERVIEW:** Introduction: FAQ's about software engineering, Professional and ethical responsibility. **Socio-Technical systems:** Emergent system properties; Systems engineering; Organizations, people and computer systems; Legacy systems.  **CRITICAL SYSTEMS, SOFTWARE PROCESSES: Critical Systems:** A simple safety-critical system; System dependability; Availability and reliability. **Software Processes**: Models, Process iteration, Process activities; The Rational Unified Process; Computer-Aided Software Engineering. | | | | |
| **Revised Bloom’s**  **Taxonomy Level** |  | | | |
| **UNIT- II (13 hours)** | | | | |
| **REQUIREMENTS:** Software Requirements: Functional and Non-functional requirements; User requirements; System requirements; The software requirements document. **Requirements Engineering Processes:** Feasibility studies; Requirements elicitation and analysis; Requirements validation; Requirements management.  **SYSTEM MODELS:** System Models: Context models; Behavioral models; Data models; Object models; Structured methods. **SOFTWARE DESIGN:** Architectural Design, System organization; Modular decomposition styles; Control styles. | | | | |
| **Revised Bloom’s**  **Taxonomy Level** |  | | | |
| **UNIT- III (13 hours)** | | | | |
| **OBJECT-ORIENTED DESIGN**: An Object-Oriented design process; Design evolution.**DEVELOPMENT:** Rapid Software Development: Agile methods; Extreme programming; Rapid application development.  **Software Evolution**: Program evolution dynamics; Software maintenance; Evolution processes; Legacy system evolution.**VERIFICATION AND VALIDATION:** Verification and Validation: Planning; Software inspections; Automated static analysis; Verification and formal methods. | | | | |
| **Revised Bloom’s**  **Taxonomy Level** |  | | | |
| **UNIT- IV (13 hours)** | | | | |
| **SOFTWARE TESTING:** System testing; Component testing; Test case design; Test automation. **PROJECT MANAGEMENT:** Project Management: Management activities; Project planning; Project scheduling; Risk management. **MANAGING PEOPLE:** Managing groups; The People Capability Maturity Model;  **SOFTWARE COST ESTIMATION:** Productivity. **DESIGNING AND DOCUMENTING SOFTWARE ARCHITECTURE:** Architecture in the life cycle; designing the architecture; Forming the team structure; Creating a skeletal system. | | | | |
| **Revised Bloom’s**  **Taxonomy Level** |  | | | |
|  | | | | |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course outcomes:**  At the end of the course the student will be able to:   * CO1: Explain the existing concepts, models and techniques used in the software development. * CO2: Write software requirement specification based on the formal specifications for software systems. * CO3: Design and develop different components of the software product using standard models. * CO4: Verify and validate the individual components and the whole software product using different testing tools. * CO5: Demonstrate the management of people, project and software quality during the software development. | | | | | | | | | |
|  | | | | | | | | | |
| **Sl No** | | **Title of the Book** | | **Name of the Author/s** | | **Name of the Publisher** | | **Edition and Year** | |
| **Textbooks** | | | | | | | | | |
| 1 | | Software Engineering | | Ian Somerville | | Pearson Education | | 8th Edition, 2007 | |
| 2 | | Software Architecture in Practice | | Len Bass, Paul Clements, Rick Kazman | | Pearson Education | | 2nd Edition, 2003 | |
| **Reference Books** | | | | | | | | | |
| 1 | | Software Engineering: A Practitioners Approach | | Roger S. Pressman | | McGraw-Hill | | 6th /7th Edition, 2007 | |
| 2 | | Software Engineering Theory and Practice | | Shari Lawrence Pfleeger, Joanne M. Atlee | | Pearson Education | | 3rd Edition, 2006 | |
| 3 | | Software Engineering Principles and Practice | | Waman S Jawadekar | | Tata McGraw-Hill | | 1st Edition, 2004 | |
| 4 | | Software Engineering | | Ian Somerville | | Pearson Education | | 10th Edition, 2018 | |
|  | | | | | | | | | |
| **Web links and Video Lectures:**   1. [http://nptel.ac.in/courses/106/101/106101061/](http://nptel.ac.in/courses/106/101/1061015061/) 2. <http://nptel.ac.in/courses/106/105/106105087/> 3. <http://nptel.ac.in/courses/106/105/106105182/> 4. <http://uml.org> 5. VTU EDUSAT PROGRAMME | | | | | | | | | |
|  | | | | | | | | | |
|  | | | | | | | | | |
| **B.E (COMPUTER SCIENCE AND ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**  **SEMESTER – VI** | | | | | | | | |
| MANAGEMENT AND ENTREPRENEURSHIP | | | | | | | | |
| Course Code | | | | **UCS654H** | | CIE Marks | | 50 |
| Teaching Hours /Week (L:T:P) | | | | (3:0:0) | | SEE Marks | | 50 |
| Credits | | | | 03 | | Hours | | 40 |
| **Course objectives:**   * To have insight into the fundamentals of management and entrepreneurship that includes the different types, roles and functions played by the managers / entrepreneurs at different levels etc. * To have proficiency in managing the activities effectively and efficiently and to be a successful entrepreneur. | | | | | | | | |
| **UNIT - I (10 hours)** | | | | | | | | |
| **Management:** Introduction - Meaning - nature and characteristics of Management, Scope and Functional areas of management - Management as art or science, art or profession - Management & Administration - Roles of Management, Levels of Management, Development of Management Thought - early management approaches - Modem management approaches  **Planning:** Nature, importance and purpose of planning process objectives - Types of plans (meaning only) - Decision making, Importance of planning - steps in planning & planning premises - Hierarchy of plans | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | |  | | | | | | |
| **UNIT- II (10 hours)** | | | | | | | | |
| **Organizing and staffing:** Nature and purpose of organization, Principles of organization – Typesof organization-Departmentation Committees-Centralization Vs Decentralization of authority and responsibility - Span of control - MBO and MBE Nature and importance of staffing-- :Process of Selection & Recruitment  **Directing:** Meaning and nature of directing Leadership styles, Motivation, Theories, Communication - Meaning and importance - coordination, meaning and importance and Techniques of coordination. **Controlling:** Meaning and steps in controlling - Essentials of a sound control system - Methods of establishing control | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | |  | | | | | | |
| **UNIT- III (10 hours)** | | | | | | | | |
| **Entrepreneur:** Meaning of Entrepreneur; Evolution of .the Concept; Functions of an Entrepreneur, Types of Entrepreneur, Entrepreneur - an emerging. Class. **Concept of Entrepreneurship** - Evolution of Entrepreneurship, Development of Entrepreneurship; Stages in entrepreneurial process; Role of entrepreneurs in Economic Development. Entrepreneurship in India; Entrepreneurship - its Barriers  **Industrial ownership:** Definition and meaning of Partnership, Characteristics of Partnership, Kinds of Partners, Partnership Agreement or Partnership Deed, Registration of Partnership Firm, Rights, Duties and Liabilities of Partners, Advantages and Disadvantages of Partnership, Sole proprietorship, Features, Scope Advantages and Disadvantages of Sole Proprietorship | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | |  | | | | | | |
| **UNIT- IV (10 hours)** | | | | | | | | |
| **Small scale industries:** Definition; Characteristics; Need and rationale; Objectives; Scope; roleof SSI in Economic Development. Advantages of SSI, Steps to start and SSI - Government policy towards SSI; Different Policies of SSI; Government Support for SSI during 5 year plans  **Impact of Liberalization, Privatization, Globalization on SSI:** Effect of WTO/GA TT Supporting Agencies of Government for SSI, Meaning, Nature of support; Objectives; Functions; Types of Help; Ancillary Industry and Tiny Industry Institutional support: Different Schemes; TECKSOK; KIADB; KSSIDC; KSIMC; DIC SingleWindow Agency; SISI; NSIC; SIDBI; KSFC | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | |  | | | | | | |
|  | | | | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Course outcomes:**  At the end of the course the student will be able to:   * CO1. Explain the different levels of management along with the different types of managers, their roles and functions etc. * CO2. Plan and organize the activities required to complete the project. * CO3. Create, motivate and manage groups/committees to carry out the assigned tasks. * CO4. Explain the fundamentals of entrepreneurship and its development process. * CO5. Establish Small Scale Industries using various types of supporting agencies and financing available for an entrepreneur. | | | | | |
|  | | | | | |
| **Sl No** | **Title of the Book** | | **Name of the Author/s** | **Name of the Publisher** | **Edition and Year** |
| **Textbooks** | | | | | |
| 1. | Principles of Management | | P. C. Tripathi,  P.N. Reddy | Tata McGraw Hill | 5th Edition, 2012 |
| 2. | Dynamics of Entrepreneurial Development & Management | | Vasant Desai | Himalaya PublishingHouse | 4th Edition, 2001 |
| **Reference Books** | | | | | |
| 1. | Management Fundamentals - Concepts, Application, Skill Development | Robert Lusier | | Thomson/South-Western | 5th Edition, 2012 |
| 2. | Entrepreneurship Development | S. S. Khanka | | S. Chand & Co. New Delhi. | 1st Revised Edition, 1999 |
| 3. | Management | Stephen Robbins | | Pearson Education/PHI | 17th Edition, 2003 |
|  | | | | | |
| **Web links and Video Lectures:**   1. <https://nptel.ac.in/courses/110/106/110106145/> 2. <https://nptel.ac.in/courses/110/105/110105146/> 3. <https://nptel.ac.in/courses/110/105/110105147/> 4. <https://nptel.ac.in/courses/110/106/110106141/> 5. <https://nptel.ac.in/courses/110/106/110106134/> 6. VTU EDUSAT PROGRAMME | | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **B.E (COMPUTER SCIENCE AND ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**  **SEMESTER – VI** | | | |
| **COMPUTER GRAPHICS LABORATORY** | | | |
| Course Code | **UCS655L** | CIE Marks | 50 |
| TeachingHours/Week (L:T:P) | (2:0:0) | SEE Marks | 50 |
| Credits | 1 | Hours/week | 2 |
| **Course objectives:**   * Have insight into graphics application andalgorithmic development of graphical images and pictures. * Have proficiency in developing graphics software for real time applications. | | | |
|  | | | |
| **Part-A**   1. Write OpenGL program to implement Bresenham’s line drawing algorithm. 2. Write OpenGL program to implement midpoint circle drawing algorithm. 3. Implement OpenGL program to draw bar chart and pie chart. 4. Write the following interactive OpenGL program   i) Draw a house using mouse to select two end point positions for straight line  ii) Display string “WEL TO BEC” on display window accepted from keyboard   1. Implement interactive animation programs.  * Kite flying * Rotating wheel * Moving car  1. Program to recursively subdivide a triangle to form 2D Sierpinski gasket. The number of recursive steps is to be specified by the user. 2. Program to draw a cube and spin it using OpenGL transformation matrices. 3. Program to create a house like figure and rotate it about a given fixed point using OpenGL functions. 4. Program to implement the Cohen- Sutherland line-clipping algorithm. 5. Program to create a cylinder and a parallelepiped by extruding a circle and quadrilateral respectively. Allow the user to specify the circle and the quadrilateral   **Part- B**  Develop a suitable graphics package to implement the skills learnt in the theory and the exercises indicated in Part A. Use the OpenGL. | | | |
| |  |  | | --- | --- | | **Revised Bloom’s**  **Taxonomy Level** | L3: Applying L4: AnalysingL5: Evaluating L6: Creating | | | | |
| **Course outcomes:**  At the end of the course the student will be able to:   * CO1: Draw the basic geometrical using OpenGL built in functions * CO2: Execute the program to implement fundamental graphics algorithms * CO3: Develop the programs to create animation of objects using graphics functions and Develop graphics applications using OpenGL programming tool. | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **MINI PROJECT** | | | | | |
| **Sub Code** | **:** | **UCS656P** | **Credits** | **:** | **03** |
| **Hours/Week** | **:** | **03** | **CIE Marks** | **:** | **50** |
| **Total Hours** | **:** | **-** | **SEE Marks** | **:** | **50** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Course objectives:**

* Have insight into current state of art and trends in their area of interest and problem defined.
* To have proficiency in design , implementation of different components using appropriate tools

Based on the ability/abilities of the student/s and recommendations of the mentor, a single discipline or a multidisciplinary Mini- project can be assigned to an individual student or to a group having not more than 4 students. The mentor shall monitor progress of the student/s continuously. The student/s is/are required topresent the progress of the Mini Project work during the semester as per the scheduleprovided by the Department Project Coordinator.

**COURSE OUTCOMES**

After completion of the Mini Project the student is able to

CO1 Develop the ability to solve real life problems related to software development.

CO2 Identify the issues and challenges in the domain.

CO3 Apply the knowledge and techniques learnt in theoretical classes.

CO4 Explain the deeper understanding in specific functional areas of the real problems.

CO5 Explore career opportunities in their areas of interest.

**CIE for Mini-Project:**

(i) **Single discipline**: The CIE marks shall be awarded by a committee consisting of the Head of the Department and two senior faculty members of the Department, one of whom shall be the Guide. The CIE marks awarded for the Mini-project work, shall be based on the evaluation of project report, project presentation skill and question and answer session in the ratio 50:25:25.The marks awarded for the project report shall be the same for all the batch mates.

(ii) **Interdisciplinary**: Continuous Internal Evaluation shall be group wise at the college level with the participation of all the guides of the college. The CIE marks awarded for the Mini-project, shall be based on the evaluation of project report, project presentation skill and question and answer session in the ratio 50:25:25.The marks awarded for the project report shall be the same for all the batch mates.

**SEE for Mini-Project:**

(i) **Single discipline**: Contribution to the Mini-project and the performance of each group member shall be assessed individually in the semester end examination (SEE) conducted at the department. (ii) **Interdisciplinary**: Contribution to the Mini-project and the performance of each group member shall be assessed individually in semester end examination (SEE) conducted separately at the departments to which the student/s belongs to.

**Scheme of Evaluation for Mini Project**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl.No.** | **Course Component** | **CIE Evaluation**  **(Max. 50 Marks)** | **SEE Evaluation**  **(Max. 50 Marks)** |
| 1 | Mini Project | Respective Guide  (Project Report, Project Presentation Skill, Interaction in the ratio of 50:25:25) | (Project Evaluation: 30 Marks and Presentation : 20 Marks)  Conducted by Departmental Committee consisting of   1. HOD/Nominee 2. Project Coordinator/Guide 3. Examiner |
| **Total Marks** | | | **100** |

**Rubrics for CIE Evaluation**

The following percentage of weightage is assigned to the student based on the performance in the CIE Evaluation

|  |  |  |
| --- | --- | --- |
| **Sl.No.** | **Performance** | **Percentage of Weightage** |
| 1 | Excellent | 91 to 100 |
| 2 | Very Good | 81 to 90 |
| 3 | Good | 71 to 80 |
| 4 | Moderate | 61 to 70 |
| 5 | Poor | 40 to 60 |

**7thSemester B.E. Computer Science & Engg**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Sl.No** | **Subject Code** | **Subjects** | **Hrs/Week** | | | **C** |  |  |  |
| **L** | **T** | **P** | **CIE** | **SEE** | **Total** |
|  | UCS751C | Web Technologies | 3 | 2 | 0 | 4 | 50 | 50 | 100 |
|  | UCS752C | Compiler Design | 3 | 2 | 0 | 4 | 50 | 50 | 100 |
|  |  | Elective IV | 3 | 0 | 0 | 3 | 50 | 50 | 100 |
|  |  | Open Elective II | 3 | 0 | 0 | 3 | 50 | 50 | 100 |
|  | UCS753L | Web Technologies Lab | 0 | 0 | 2 | 1 | 50 | 50 | 100 |
|  | UCS754L | Computer Networks Lab | 0 | 0 | 2 | 1 | 50 | 50 | 100 |
|  | UCS755P | Project Phase I | 0 | 0 | 10 | 5 | 50 | 50 | 100 |
|  | UCS756I | Internship | 0 | 0 | 4 | 2 | 50 | 50 | 100 |
|  |  |  | **12** | **4** | **18** | **23** | 400 | 400 | 800 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **B.E (*Computer Science and Engineering*)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**  **SEMESTER -VII** | | | | | | | | | | |
| **Web Technologies** | | | | | | | | | | |
| **Course Code** | | UCS751C | | | | | | **CIE Marks** | 50 | |
| **TeachingHours/Week (L:T:P)** | | (3:2:0) | | | | | | **SEE Marks** | 50 | |
| **Credits** | | 04 | | | | | | **Hours** | 40 +24=64 | |
| **Course Objectives**   1. Have insight into World Wide, HTML/XHTML, Java Script,PHP. 2. Have proficiency in design of web applications which will work with database. | | | | | | | | | | |
| **Unit -1 (10 hours+6 hours Tutorial)** | | | | | | | | | | |
| **Fundamentals:** A Brief Introduction to the Internet,The World WideWeb,Web Browsers,Web Servers, Uniform Resource Locators,Multipurpose Internet MailExtensions ,The Hypertext Transfer Protocol, Security, The Web Programmer’sToolbox. Introduction to HTML/XHTML: Origins and Evolution of HTML and XHTML, Basic Syntax,Standard HTML Document Structure,Basic Text Markup, Images, HypertextLinks, Lists; Tables,Forms The: Audio Element,TheVideoElement,OrganizationElements,TheTimeElement,SyntacticDifferencesbetweenHTMLandXHTML. **Cascading Style Sheets:** Introduction, Levels of Style Sheets,Style SpecificationFormats, Selector Forms, Property-Value Forms, Font Properties, ListProperties, Alignment of Text,Color: The BoxModel, BackgroundImages, The span and divTags,ConflictResolution. | | | | | | | | | | |
| **Revised Bloom’sTaxonomyLevel** | | | | Remembering, | | | | | | |
| **UNIT- II (10 hours+6 hours Tutorial)** | | | | | | | | | | |
| **The Basics of javascript:** Overview ofjavascript, Object Orientation andjavascript,General Syntactic Characteristics, Primitives, Operations, and Expressions, Screen Output and Keyboard Input, Control, Statements, Object Creation andModificationArrays, Functions, AnExample, Constructors,Pattern Matching Using RegularExpressions,AnotherExample.  **JavaScript and HTML Documents:**The java script Execution Environment**,**The Document Object Model**,** Element Access injavascript**,** Events and Event Handling.Handling Events from Body Elements**,** Handling Events from Button Elements Handling EventsfromTextBoxand Password ,Elements: The DOM 2 EventModel, The canvas Element,The navigatorObject,DOM Tree Traversal and Modification. | | | | | | | | | | |
| **Revised Bloom’sTaxonomy Level** | | | Remembering, | | | | | | | |
| **UNIT- III (10 hours+6 hours Tutorial)** | | | | | | | | | | |
| **Dynamic Documents with Java Script:** Introduction,PositioningElements, Moving Elements, Element Visibility, Changing Colors and Fonts, Dynamic Content,Stacking Elements, Locating the Mouse Cursor, Reacting to a Mouse Click, Slow Movement ofElements, Dragging and DroppingElements.  Introduction to jQuery: [WhyjQuery?](#_bookmark1774) [Includingj Query](#_bookmark1776), [jQuery Syntax](#_bookmark1789)[A SimpleExample](#_bookmark1792), [Avoiding Library Conflicts](#_bookmark1796), [Selectors](#_bookmark1800),[Handling Events](#_bookmark1814),[Waiting Until the DocumentIs Ready](#_bookmark1818), [Event Functions and](#_bookmark1822)  [Properties](#_bookmark1822), [Special Effects](#_bookmark1855),[Manipulatingthe DOM](#_bookmark1885) [Dynamically Applying Classes](#_bookmark1902), [Modifying Dimensions](#_bookmark1904), [DOM Traversal](#_bookmark1915), [Usingj Query Without Selectors](#_bookmark1940),[UsingAsynchronous Communication](#_bookmark1947) [Plug-ins](#_bookmark1955). | | | | | | | | | | |
| **Revised Bloom’sTaxonomy Level** | | | | | | Remembering,L4-Analyze | | | | |
| **UNIT- IV (10 hours+6 hours Tutorial))** | | | | | | | | | | |
| **Introduction to XML:** Introduction, Uses of XML, The Syntax of XML, XML Document Structure, Namespaces, XML Schemas, Displaying Raw XML Documents ,Displaying XML Documents withCSS, XSLT Style Sheets, XMLProcessors,Web Services.  **Introduction to PHP:** Origins and Uses of PHP,Overview of PHP, General Syntactic Characteristics, Primitives, Operations, and Expressions,Output, Control Statements, Arrays, Functions, Pattern Matching, Form Handling,Cookies, Session Tracking.  **Database Access through theWeb:** Database Access with PHP and MySQL. | | | | | | | | | | |
| **Revised Bloom’sTaxonomy Level** | | | | | | Remembering, | | | | |
| **Course outcomes:**  At the end of the course the student will be able to:  CO1: Explain the basics of World Wide Web.  CO2: Implement web concepts  using  different tools like HTML/XHTML/CSS/JavaScript/XML/XSLT.  CO3: Design web applications using client-side Java Scripts.  CO4: Implement web applications using server –side  PHP.  CO5: Develop web application for real world problem. | | | | | | | | | | | |
| **Sl No** | | **Title of the Book** | | | | **Name of the Author/s** | | **Name of the Publisher** | | | **Edition and Year** |
| **Textbooks** | | | | | | | | | | | |
| 1 | | Programming the World Wide Web | | | | Robert W. Sebesta | | Pearson Education | | | 8th Edition, 2014 |
| **Reference Books** | | | | | | | | | | | |
| 1 | | Learning PHP, MySQL & JavaScript | | | | Robin Nixon | | O’Reilly Publications | | | 5thEdition, 2015 |
| 2 | | Internet & World Wide Web How to program | | | | M. Deitel, P.J.Deitel,  A. B. Goldberg | | Pearson Education / PHI | | | 3rd Edition, 2004 |
| 3 | | Web Programming Building Internet Applications | | | | Chris Bates | | Wiley India | | | 3rd Edition,2006 |
| **Web links and Video Lectures:**  1. http://www.w3schools.com  2. http://nptel.iitm.ac.in. | | | | | | | | | | | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **B.E (COMPUTER SCIENCE AND ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**  **SEMESTER – VII** | | | | | | | | |
| **Compiler Design** | | | | | | | | |
| Course Code: | | | | UCS752C | CIE Marks | | 50 | |
| Teaching Hours/Week (L:T:P) | | | | (4:0:0) | SEE Marks | | 50 | |
| Credits | | | | 04 | Hours | | 48 | |
| **Course objectives:**  1. To understand the basic principles of compiler design, its various constituent parts, algorithms and data structures required to be used in the compiler.  **2.** To provide the skills needed for building compilers for various situations that one may encounter in a career in Computer Science. | | | | | | | | |
| **Unit -1 (12 hours)** | | | | | | | | |
| **Introduction, lexical analysis:** Language processors; The structure of a Compiler; Lexical analysis: The Role of Lexical Analyzer; Input Buffering; Specifications of Tokens; Recognition of Tokens. Lexical Analyzer generator  **Syntax analysis – 1:** Introduction; Context-free Grammars; Writing a Grammar; Top-down Parsing. | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | | Remembering,L3 –Applying L4-Analysis | | | | | |
| **UNIT- II (13 hours)** | | | | | | | | |
| **Syntax analysis – 2:** Bottom-up Parsing; Introduction to LR Parsing: Simple LR, Parser Generators.  **Syntax-directed translation:** Syntax-Directed definitions; Evaluation order for SDDs; Applications of Syntax-directed translation; Syntax-directed translation schemes.  **Sorting, Algorithms for Generating Combinatorial Objects.** | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | | Remembering,L3 –Applying L4-Analysis | | | | | |
| **UNIT- III (12 hours)** | | | | | | | | |
| **INTERMEDIATE CODE GENERATION:** Variants of syntax trees; Three-address code; Types and declarations; Translation of expressions; Type checking; Control flow; Backpatching. | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | | Remembering,L3 –Applying L4-Analysis | | | | | |
| **UNIT- IV (13 hours)** | | | | | | | | |
| **RUN-TIME ENVIRONMENTS:** Storage Organization; Stack allocation of space, Access to non-local data on the stack; Heap management;  **CODE GENERATION:** Issues in the design of Code Generator; The Target language; Addresses in the target code; Basic blocks and Flow graphs; Optimization of basic blocks | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | | Remembering,L3 –Applying L4-Analysis | | | | | |
| **Course outcomes:**  At the end of the course the student will be able to:  CO 1: Demonstrate the understanding of  different phases of Compiler Design.  CO 2: Explain the working of Lexical Analysis  CO 3: Apply the concept of Syntax Analysis ,Syntax directed translation and intermediate code generation  CO 4: Demonstrate the knowledge of Runtime Environments & Code Optimization  techniques.  CO 5: Design & implement a scanner and a parser using LEX & YACC tools for simple problems. | | | | | | | |
| **Sl No** | **Title of the Book** | **Name of the Author/s** | | | **Name of the Publisher** | | **Edition and Year** |
| **Textbooks** | | | | | | | |
| **1** | **Compilers- Principles, Techniques and Tools** | Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D Ullman, Chapter 1 : 1.1 -1.2, chapter 3:3.1 to 3.5, Chapter 4: 4.1-4.6, 4.9.1-4.9.3, Chapter 5:5.1-5.4.5, Chapter 6:6.1-6.5.2, 6.6, 6.7-6.7.3, Chapter 7:7.1-7.4 Chapter 8:8.1-8.6) | | | Addison-Wesley. | | 2nd Edition, 2007 |
| **Reference Books** | | | | | | | |
| **1** | **Crafting a Compiler with C** | Charles N. Fischer, Richard J. leBlanc, Jr, | | | Pearson Education | | 1991 |
| **2** | **Modern Compiler Implementation in C** | Andrew W Apple, | | | Cambridge University Press. | | 1998 |
| **3** | **Compiler Construction Principles & Practice** | Kenneth C  Louden | | | Thomson Education. | | 1997 |
| **4** | L**ex &Yacc** | John Levine, Doug Brown, Tony Mason | | | O'Reilly Media, 2nd Edition | | 1992 |
| **Web links and Video Lectures:**  NPTEL course on Principles of Compiler Design :  https://nptel.ac.in/courses/106/108/106108113/  3. VTU EDUSAT PROGRAMME – 20 | | | | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **B.E (*Computer Science and Engineering*)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**  **SEMESTER -VII** | | | | |
| **Web Technologies Laboratory** | | | | |
| **Course Code** | | UCS753L | **CIE Marks** | 50 |
| **Teaching Hours/Week (L:T:P)** | | (0:0:2) | **SEE Marks** | 50 |
| **Credits** | | 01 | **Exam Hours** | 03 |
| Course Objectives:  1. Have insight into World Wide, HTML/XHTML, Java Script,jQuery,PHP. 2. Have proficiency in design of web applications which will work with database. | | | | |
| **Sl. No.** | | **Problem Statement** | | | |
| 1 | | Design and develop static web page using HTML to demonstrate tables, different forms of hypertext links and frames. | | | |
| 2 | | Design and develop web page to demonstrate CSS (Use different fontstyles, set background image for both the page and single elements onpage, Control the repetition of image with background-repeatproperty, define style for links as a:link, a:active, a:hover,a:visited, Add customized cursors forlinks, Work withlayers) | | | |
| 3 | | Develop web page to demonstrate Form validation using JavaScript. | | | |
| 4 | | Develop dynamic web page to demonstrate Positioning Elements, Moving Elements, Element Visibility, Changing Colors and Fonts, Dynamic Content, Stacking Elements, Locating the Mouse Cursor, reacting to a Mouse Click. | | | |
| 5 | | Program to demonstrate jQUERY. | | | |
| 6 | | Development of XML file and write a DTD to validate the XML file and display the details in a table using XSL. | | | |
| 7 | | PHP program to demonstrate Cookie creation, display and deletion. | | | |
| 8 | | PHP program to demonstrate session. | | | |
| 9 | | PHP program to perform CRUD operation on database. | | | |
| 10 | | PHP Program to validate the input data and store the acquired data to database. And also display the status of execution of operation. | | | |
| **Course outcomes:**  At the end of the course the student will be able to:  CO1: Create and Manage static web pages for given scenario  CO2: Design web applications using client-side Java Scripts.  CO3: Write XML/XSLT and jQuery programs for solving real-world problems.  CO4: Implement web applications using server –side  PHP programming and database connectivity.  CO5: Develop web applications with sessions. | | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sl No** | **Title of the Book** | **Name of the Author/s** | **Name of the Publisher** | **Edition and Year** |
| **Textbooks** | | | | |
| **1** | Programming the World Wide Web | Robert W. Sebesta | Pearson Education | 8th Edition, 2014 |
| **References** | | | | |
| **1** | Learning PHP, MySQL & JavaScript | Robin Nixon. | O'Reilly Media | 5th edition. May 2018 |
| **Web links and Video Lectures:**  1. http://www.w3schools.com  2. http://nptel.iitm.ac.in. | | | | |

|  |
| --- |
| **B.E (COMPUTER SCIENCE AND ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**  **SEMESTER – VII** |
| **Computer Networks Lab** |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | **Course Code:** | **UCS754L** | **CIE Marks** | **50** | | **Teaching Hours** | **(0:0:2)** | **SEE Marks** | **50** | | **Credits** | **01** | **Exam Hours:** | **03** |   **Course objectives:**   1. To have insights into design and implementation of current topics in computer networks such as congestion control, network security and routing. 2. To have proficiency in developing network applications.   **PART –A**  **Simulation Exercises**  **Introduction Part**  Introduce students to network simulation through the Network simulation Package, Create a simple network model with multiple scenarios, Collect statistics on network performance through the use of simulator tools, Analyze and draw conclusion on network performance   1. Simulate four nodes point-to-point network and study how the loss, utilization and transmission of wireless LAN (IEEE 802.11b) network varies as the distance between access point and wireless nodes. 2. Simulate point-to-point network which consists of 4 to 6 nodes and study network performance analysis of different scheduling technique like First In Out (FIFO), Priority, Round Robin, Weight Fair Queue (WFQ) using NetSim. 3. Simulate and study the throughputs of slow start, Congestion avoidance (also known as Old Tahoe) and First Retransmit (also known as Tahoe), Congestion Control Algorithms during client-server TCP downloads. 4. Create a network topology which consists of six nodes, simulate and study the working and routing table formation of Interior Routing Protocol i.e. Routing Information Protocol (RIP) and Open Shortest Path First (OSPF).   **PART – B**  **Implement the following in C/C++:**   1. Write a program for error detecting code using CRC-CCITT (16 bit) 2. Write a program for hamming code generation for error detection and correction. 3. Write a program for distance vector algorithm to find suitable path for transmission. 4. Write a program for congestion control using leaky bucket algorithm. 5. Write a C program to develop a DNS client server to resolve the given hostname. 6. Write a client-server application for chat using UDP. 7. 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 to the requested file if present. 8. Write a program for simple RSA algorithm to encrypt and decrypt the data.   **Note:**  Student is required to solve one problem from PART-A and one problem from PART-B. The questions are allotted based on lots. Both questions carry equal marks. | | | | | | |
| **Course outcomes:**   1. Simulate the network with different configurations to measure the performance parameters. 2. Implement the data link, network layerand application layer protocols. 3. Analyze routing algorithm to find the suitable path for transmission and control of flow rate. 4. Enable communication between the peers using TCP/IP and UDP sockets. | | | | | | | |
| **Sl No** | | | **Title of the Book** | **Name of the Author/s** | **Name of the Publisher** | | **Edition and Year** |
| **Textbooks** | | | | | | | |
| **1** | | Data Communications and Networking. | | Behrouz A. Forouzan | Tata McGraw-Hill | 4th Edition, 2017 | |
| **2** | | TCP/IP Sockets in C – A Practical Guide for Programmers | | Michael J Donahoo  Kenneth L Calvert | Morgan Kaufmann | 2nd Edition, 2009 | |
| **Reference Books** | | | | | | | |
| **1** | | Computer Networking-A top-down approach featuring the Internet | | James F. Kurose, Keith W. Ross | Pearson Education | 3rd Edition, 2018 | |
| **2** | | Data and Computer Communication | | William Stallings | Pearson Education | 8th Edition, 2016 | |

|  |  |  |  |
| --- | --- | --- | --- |
| **B.E (COMPUTER SCIENCE AND ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)** | | | |
| **Project Phase I** | | | |
| Course Code | **UCS755P** | CIE Marks | 50 |
| Teaching Hours /Week (L: T:P) | (0:0:10) | SEE Marks | 50 |
| Credits | 05 | Hours | 10/Week |
| **Course Objectives**   1. To develop the capacity of students in correlating theoretical knowledge into practical systems either to perform creative works or to perform analysis and hence to suggest solutions to problems, pertaining to civil engineering domain. 2. Foster collaborative learning skills. 3. Develop self-directed inquiry and life-long skills. 4. To enhance the communication skills of the students by providing opportunities to discuss in groups and to present their observations, findings and report in formal reviews both in oral and written format. | | | |
| Engineering projects are the means of design and implementation of assimilated technical knowledge towards the betterment of the society and market needs. A systematic and organized approach towards the engineering project is to be taken up by the department.  **Project Phase I**  Students who complete 6th semester are informed to come up with project ideas when entering in to7thsemester. A list of project ideas generated collectively by the faculty are also displayed during the first week of 7th semester. A project coordinator is designated by the Head of the Department. A batch of maximum 4 students forms a project group as per affiliated University guidelines.  **Project phase I** is carried during 7thsemester of BE.   1. Students are informed to submit the synopsis, re-tune the synopsis and conduct literature survey/review to consolidate the idea and technical viabilities, in consultation with the project guide. 2. Students present the progress of the work periodically. 3. The internal evaluation panel (consisting of Head, Project coordinator and Guide(s) monitors the progress.   The students work is periodically checked and evaluated. | | | |
| **Course Outcomes**  On completion of this course, the students will be able to   1. Submit a project synopsis comprising of the application and feasibility of the project. 2. Design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health care, safety and sustainability. 3. Work and communicate efficiently in multidisciplinary teams 4. Identify, formulate, and solve engineering problems. 5. Develop an understanding of professional and ethical responsibility. | | | |
| **Internal Evaluation for Project Phase I (CIE)**  Evaluation for the total of 50 marks is divided into Four components and is given below:   |  |  |  | | --- | --- | --- | | Component | Particulars | Marks | | A1.1 | Synopsis presentation (In the beginning of 7th semester) | 15 | | A1.2 | Review/ Mid semester evaluation (In the middle of 7thsemester | 10 | | A1.3 | 3rd Internal Evaluation (At the end of 7thsemester) | 10 | | A1.4 | Evaluation by the Guide(s) | 15 | |  | Total CIE Marks | 50 |   **SEE Evaluation for Project Phase I:** MaximumMarks:50  HOD (or nomination) +Project coordinator +Guide(s) each will evaluate for 50 marks and average of all three will be taken. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **B.E (COMPUTER SCIENCE AND ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)** | | | |
| **Internship** | | | |
| Course Code | **UCS756I** | CIE Marks | 50 |
| Teaching Hours /Week (L: T:P) | (0:0:4) | SEE Marks | 50 |
| Credits | 02 | Hours | 04/Week |
| Course Objectives:   1. To enhance and/or expand the student's knowledge of a particular area(s) of software industry. 2. To experience integration theory and practice and assess interests and abilities in their field of study. 3. Develop work habits and attitudes necessary for job success and communication, interpersonal and other critical skills for their professional career. 4. To build a record of work experience in their interested field of computer science and engineering. | | | |
| **Internship:**  Students need to meet following criteria to successfully complete the internship course.   1. **Student’s Diary/ Daily Log**   Student’s Diary and Internship Report should be submitted by the students along with attendance record and an evaluation sheet duly signed and stamped by concerned authority to the Institute immediately after the completion of the training. It will be evaluated based on the following criteria:   * Regularity in maintenance of the diary. * Adequacy & quality of information recorded. * Drawings, sketches, and data recorded. * Thought process and recording techniques used. * Organization of the information.  1. **Internship Report**   The Internship report will be evaluated based on following criteria:   * Originality. * Internship certificate from the concerned authority of the industry.. * Adequacy and purposeful writeup. * Organization, format, drawings, sketches, style, language etc. * Variety and relevance of learning experience. * Practical applications, relationships with basic theory and concepts taught in the course. | | | |
| **Course outcomes:**   1. Demonstrate the knowledge gained during the internship at the industry. 2. Exhibitabilities to use theoretical concepts in solving practical problems in their field of study. Demonstrate communication, interpersonal and other critical skills in their profession. | | | |
| **Evaluation:**  The industrial training of the students will be evaluated in three stages:   1. Evaluation by Industry. 2. Evaluation through seminar presentation 3. Viva voce at the Institute.   **Evaluation Through Seminar Presentation/Viva‐Voce at The Institute**  The student will give a seminar based on his training report, before an expert committee constituted by the concerned departmentaspernormsoftheinstitute. The evaluation will be based on the following criteria:   * + Quality of content presented.   + Proper planning for presentation.   + Effectiveness of presentation.   + Depth of knowledge and skills.   + Attendance record, daily diary, departmental reports shall also be analysed along with the Internship Report.  |  |  | | --- | --- | | **Summary of Internship Evaluation (Industry Representative)** | | | **Evaluation Criteria** | **Score from the above tables** | | Quality of Work | 10 | | Ability to Learn | 10 | | Initiative and Creativity | 10 | | Character Traits | 10 | | Dependability | 10 | | Organizational Fit | 10 | | Response to Supervision | 10 | |  | **70** | | **Internship Guide** | | | Demonstration of experience | 10 | | Report | 10 | | Presentation | 10 | |  | **30** |   Marks obtained from industry will be considered as Internship CIE for 50 marks and Marks obtained from Internal committee evaluation will be treated as SEE for 50 marks. | | | |

**8thSemester B.E. Computer Science & Engg**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Sl.No** | **Subject Code** | **Subjects** | **Hrs/Week** | | | **C** |  |  |  |
| **L** | **T** | **P** | **CIE** | **SEE** | **Total** |
|  |  | Elective V | 3 | 0 | 0 | 3 | 50 | 50 | 100 |
|  |  | Elective VI | 3 | 0 | 0 | 3 | 50 | 50 | 100 |
|  |  | Elective VII | 3 | 0 | 0 | 3 | 50 | 50 | 100 |
|  | UCS851S | Seminar | 0 | 0 | 2 | 1 | 50 | 50 | 100 |
|  | UCS852P | Project Phase II | 0 | 0 | 24 | 12 | 50 | 50 | 100 |
|  |  |  | **9** | **0** | **26** | **22** | 250 | 250 | 500 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **B.E (SCIENCE AND COMPUTER ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)**  **ELECTIVE** | | | | | | | | | |
| **Block Chain Technology** | | | | | | | | | |
| **Course Code** | | | **UCS070E** | | | | **CIE Marks** | | **50** |
| **TeachingHours/Week (L:T:P)** | | | **(3:0:0)** | | | | **SEE Marks** | | **50** |
| **Credits** | | | **03** | | | | **Hours** | | **40** |
| **Course objectives:**   1. Have insight into Block Chain Technology. 2. Have proficiency in design of applications using Block Chain Technology. | | | | | | | | | |
| **Unit -1 (10 hours)** | | | | | | | | | |
| Block chain 101: Distributed systems, History of block chain, Introduction to block chain: various technical def of block chain, Generic Elements, Features, Applications,,  Types of block chain: Public, Private, Semi-private, Side chain, Permissioned ledger, Distributed ledger, Shared, Fully private and proprietary block chains, Tokenized and tokenless block chains, Consensus block chains, CAP theorem and blockchain, Benefits and limitations of  blockchain. | | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | L1- Remembering,L2-,L3-Apply,L4-Analyze | | | | | | | |
| **UNIT- II (10 hours)** | | | | | | | | | |
| Decentralization and Cryptography:  Decentralization using block chain, Methods of decentralization, Routes to decentralization, Block chain and full ecosystem decentralization, Smart contract, Decentralized organizations.  Cryptography and Technical Foundations: Cryptographic primitives, Asymmetric cryptography, | | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | L1- Remembering,L2-,L3-Apply,L4-Analyze,L5-Evaluate | | | | | | | |
| **UNIT- III (10 hours)** | | | | | | | | | |
| Bitcoin and Alternative Coins  A: Bitcoin, Transactions life cycle, structure, types of transaction , Block chain: structure of block and header, Genesis block,bitcoin network, Wallets, Bitcoin payments: investment and buying and selling bitcoins, Bitcoin installation, Bitcoin programming and command line interface, BIPS  B: Alternative Coins  Theoretical foundations: proof of work, Difficulty adjustment and retargeting algorithms, Bitcoin limitations, | | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | L1- Remembering,L2-,L3-Apply,L4-Analyze,L5-Evaluate | | | | | | | |
| **UNIT- IV (12 hours)** | | | | | | | | | |
| Smart Contracts and Ethereum 101:  Smart Contracts: Definition, Ricardian contracts: Smart contract templates, oracles, Smart oracles, Deploying smart contracts on a block chain.  Ethereum 101: Introduction, Ethereum block chain, Elements of the Ethereum Block chain,  Precompiled contracts. | | | | | | | | | |
| **Revised Bloom’s**  **Taxonomy Level** | | L1- Remembering,L2-,L3-Apply,L4-Analyze,L5-Evaluate,L6-Create | | | | | | | |
| **Course Outcomes:**  At the end of the course, students are able to:   1. Define and Explain the fundamentals of Block chain technology. 2. Illustrate the technologies of block chain. 3. Describe the models of block chain Technology. 4. Demonstrate the Block chain Technology using Ethereum. | | | | | | | | | | |
|  | | | | | | | | | | |
| **Sl No** | | **Title of the Book** | | | **Name of the Author/s** | **Name of the Publisher** | | | **Edition and Year** | |
| **Textbooks** | | | | | | | | | | |
| 1 | | “Mastering Blockchain - Distributed ledgers, decentralization and smart contracts explained | | | Imran Bashir | Packt Publishing Ltd, | | | 2nd Edition, ISBN 978-1-78712-544-5, 2017 | |
| , | | | | | | | | | | |
|  | | | | | | | | | | |
| **Reference Books** | | | | | | | | | | |
| 1. | | Bitcoin and Cryptocurrency Technologies | | | Arvind Narayanan, Joseph Bonneau, , Edward Felten, | |  | | 2016 | |
| 2. | | Block chain Basics: A Non-Technical Introduction in 25 Steps, | | | Daniel Drescher, | | Apress, | | First  Edition, 2017 | |
| 3. | | Mastering Bitcoin: Unlocking Digital Crypto currencies | | | Andreas M. Antonopoulos | | O'Reilly  Media, | | First Edition, 2014 | |
| , , | | | | | | | | | | |
| **Web links and Video Lecture:**   1. [Introduction to Blockchain Technology and Applications - Course (nptel.ac.in)](https://onlinecourses.nptel.ac.in/noc20_cs01/preview) <https://onlinecourses.nptel.ac.in/noc20_cs01/preview> 2. [Blockchain Architecture Design and Use Cases - Course (nptel.ac.in)](https://onlinecourses.nptel.ac.in/noc19_cs63/preview)   <https://onlinecourses.nptel.ac.in/noc19_cs63/preview>  https://nptel.ac.in/courses/106104220 | | | | | | | | | | |
|  | | | | | | | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **B.E (COMPUTER SCIENCE AND ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)** | | | |
| **Seminar** | | | |
| Course Code | **UCS851S** | CIE Marks | 50 |
| Teaching Hours /Week (L: T:P) | (0:0:2) | SEE Marks | 50 |
| Credits | 01 | Hours | 02/Week |
| **Seminar:**  Seminars are used as course delivery modes to encourage students to gather current trends in technology, research literature, and self-learn topics of their interest. Seminars require students to research a technical topic, make presentations and write a detailed document on their findings individually under the guidance of faculty.  **Course outcomes (COs):**  The student is expected to:  1. Identify seminar topics based on contemporary technical, societal, and environmental issues.  2. Conduct literature survey on complex issues in the selected domain  3. Explore advanced technologies  4. Make good oral and written technical presentations | | | |
| **Evaluation:**  Seminar will be evaluated based on the following criteria:   |  |  |  | | --- | --- | --- | | **Sl. No** | **Criteria** | **Marks** | | 1. | Understand problems and select Topic from journal/transaction papers from ACM/Elsevier/ Springer/IEEE etc.. | 03 | | 2. | Societal/ environmental/ Ethical relevance of the topic | 02 | | 3. | Ability to collect required number of back ground materials | 03 | | 4. | Ability to select papers with latest technical knowledge and tools | 03 | | 5. | Preparation of slides | 05 | | 6. | Presentation | 15 | | 7. | Knowledge on the topic | 04 | | 8. | Report | 15 |   Seminar will be evaluated by conducting CIE examination for 50 marks and SEE examination for 50 marks. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **B.E (COMPUTER SCIENCE AND ENGINEERING)**  **Outcome Based Education (OBE) and Choice Based Credit System (CBCS)** | | | |
| **Project Phase II** | | | |
| Course Code | **UCS852P** | CIE Marks | 50 |
| Teaching Hours /Week (L: T:P) | (0:0:24) | SEE Marks | 50 |
| Credits | 12 | Hours | 24/Week |
| **COURSE OBJECTIVES**   1. To develop the capacity of students in correlating theoretical knowledge into practical systems either to perform creative works or to perform analysis and hence to suggest solutions to problems, pertaining to Computer engineering engineering domain. 2. Foster collaborative learning skills. 3. Develop self-directed inquiry and life-long skills. 4. To enhance the communication skills of the students by providing opportunities to discuss in groups and to present their observations, findings and report in formal reviews both in oral and written format. | | | |
| Engineering projects are the means of design and implementation of assimilated technical knowledge towards the betterment of the society and market needs. A systematic and organized approach towards the engineering project is to be taken up by the department  Project phase –II  It is carried during 8thsemesterof BE.Students are monitored as they carryout the project as per the stated schedule. The progress of project work is monitored by the panel of internal faculty. The students are made to present the work done and demonstrate modules built on regular basis and evaluationareto becarried outusingprescribed rubrics. Internal evaluation will be carried out by committee comprising of Head of department, guides of project teams and project coordinator.  **Course Outcomes**  On completion of this course, the students will be able to   1. Submit a project synopsis comprising of the application and feasibility of the project. 2. Design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health care, safety and sustainability. 3. Work and communicate efficiently in multidisciplinary teams 4. Identify, formulate, and solve engineering problems. 5. Develop an understanding of professional and ethical responsibility. | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **B1: Internal Evaluation for Project Phase II Maximum Marks:50**  **Evaluation for the total of 50 marks is divided into five components as iven below:**   |  |  |  | | --- | --- | --- | | **Component** | **Particulars** | **Marks** | | **B1.1** | **Project work progress review #1 (Beginning of 8thsemester)** | **10** | | **B1.2** | **Project work progress review #2 (Middle of 8th semester)** | **10** | | **B1.3** | **Project work progress review #3 (End of 8th semester)** | **10** | | **B1.4** | **Evaluation by the guide** | **10** | | **B1.5** | **Project Report Evaluation** | **10** | |  | **Total CIE Marks** | **50** | |
| **B2: SEE Evaluation for Project Phase II MaximumMarks:50**  HOD (or nomination) + Project coordinator + External Examiner Each will evaluate for 50 marks  and average of all three will be taken. |