NEW HORIZON
COLLEGE OF ENGINEERING
Permanently affiliated to VTU, approved by AICTE & ISO 9001:2008 certified
The Trust is a Recipient of Prestigious Rajyotsava State Award 2012 Conferred by the Government of Karnataka
Awarded Outstanding Technical Education Institute in Karnataka-2014
Ring Road, Bellandur Post, Near Marathalli, Bangalore -560 103, INDIA, Fax: +91-80-2844 0770

(An Autonomous Institution Affiliated to VTU)

Accredited by NAAC with 'A' Grade

Department of Computer Science and Engineering

Academic Year 2016-17

Third and Fourth Semesters B.E

Scheme and Syllabus

2015-19 Batch

CONTENTS

VISION

To emerge as a department of eminence in Computer Science and Engineering in serving the Information Technology Industry and the nation by empowering students with a high degree of technical and practical competence.

MISSION

- To strengthen the theoretical and practical aspects of the learning process by strongly encouraging a culture of research, innovation and hands-on learning in Computer Science and Engineering

- To encourage long-term interaction between the department and the IT industry, through the involvement of the IT industry in the design of the curriculum and its hands-on implementation

- To widen the awareness of students in professional, ethical, social and environmental dimensions by encouraging their participation in co-curricular and extracurricular activities

Program Education objectives (PEOs)

PEO1 Proficiency as computer scientists with an ability to solve a wide range of computing- related problems in industry, government, or other work environments.

PEO2 Ability to adapt quickly to new environments and technologies, assimilate new information, and work in multi-disciplinary areas with a strong focus on innovation and entrepreneurship.

PEO3 Possess the ability to think logically and the capacity to understand technical problems with computing systems and design alternative solutions.

PEO4 Possess an ability to collaborate as a team member and team leader to affect technical solutions for computing systems, providing improved function and outcomes.

PEO to Mission Statement Mapping

Mission Statements	PEO1	PEO2	PEO3	PEO4
To educate graduates and research scholars to be successful, ethical, and effective problem-solvers and life-long learners.	3	-	2	-
Produce versatile Computer Science graduates infused not only with technical skills, but also with innovative and entrepreneurial skills.	-	3	-	-
Prepare graduates for successful careers in Software Industry.	3	3	3	3
Provide a great work and learning environment and treat each other with respect and dignity.	-	-	2	3
To prepare graduates well enough to function as professional computer scientists and computer engineers.	-	3	-	-

Correlation: 3- High, 2-Medium, 1-Low

Program Outcomes (PO) with Graduate Attributes

	Graduate Attributes	Program Outcomes (POs)
1	Engineering Knowledge	**PO1:** The basic knowledge of Mathematics, Science and Engineering.
2	Problem analysis	**PO2:** An Ability to analyse, formulate and solve engineering problems.
3	Design and Development of Solutions	**PO3:** An Ability to design system, component or product and develop interfaces among subsystems of computing.
4	Investigation of Problem	**PO4:** An Ability to identify, formulate and analyze complex engineering problem and research literature through core subjects of Computer Science.
5	Modern Tool usage	**PO5:** An Ability to use modern engineering tools and equipment for computing practice.
6	Engineer and society	**PO6:** An Ability to assess societal, health, cultural, safety and legal issues in context of professional practice in Computer Science & Engineering.
7	Environment and sustainability	**PO7:** The broad education to understand the impact of engineering solution in a global, economic, environmental and societal context.
8	Ethics	**PO8:** An understanding of professional and ethical responsibility.
9	Individual & team work	**PO9:** An Ability to work both as individual and team player in achieving a common goal.
10	Communication	**PO10:** To communicate effectively both in written and oral formats with wide range of audiences.
11	Lifelong learning	**PO11:** Knowledge of contemporary issues, Management and Finance.
12	Project management and finance	**PO12:** An Ability to recognize the need and thereby to engage in independent and life-long learning for continued professional and career advancement.

Mapping of POs TO PEOs

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
PEO1	3	3	3	-	-	-	-	-	-	-	-	-
PEO2	-	-	-	-	-	-	-	-	-	-	-	-
PEO3	-	-	3	3	-	-	-	-	-	-	3	-
PEO4	-	-	-	-	-	-	-	-	3	-	-	-

Correlation: 3- High, 2-Medium, 1-Low

New Horizon College of Engineering
Department of Computer Science and Engineering
Scheme of Third Semester B.E Program

Sl. No	Course Code	Course	Credit Distribution				Overall Credits	Contact Hours weekly- Theory	Contact Hours weekly- (Lab)	Marks		
			L	P	T	S				CIE	SEE	TOTAL
1	16MAT31	Engineering Mathematics-III	4	0	1	0	5	6	0	50	50	100
2	16HSS321	Introduction to Economics	2	0	0	1	3	2	0	50	50	100
	16HSS322	Soft Skills for Engineers										
3	16CSE33	Analog and Digital Electronics	3	2	0	0	5	3	4	75	75	150
4	16CSE34	Data Structures using C	3	2	0	0	5	3	4	75	75	150
5	16CSE35	UNIX & SHELL Programming	3	2	0	0	4	3	4	75	75	150
6	16CSE36	Mini Project	0	2	0	0	2	0	4	50	50	100
Total							**24**	**17**	**16**	**375**	**375**	**750**

New Horizon College of Engineering
Department of Computer Science and Engineering
Fourth Semester B.E Program-Scheme

Sl. No	Course Code	Course	Credit Distribution				Overall Credits	Contact Hours weekly Theory	Contact Hours weekly (Lab)	Marks		
			L	P	T	S				CIE	SEE	TOTAL
1	16MAT41	Engineering Mathematics-IV	4	0	1	0	5	6	0	50	50	100
2	16HSS421	Introduction to Economics	2	0	0	1	3	2	0	50	50	100
	16HSS422	SS for Engineers										
3	16CSE43	Object Oriented Programming with C++	3	2	0	0	5	3	4	75	75	150
4	16CSE44	Microprocessor	3	2	0	0	5	3	4	75	75	150
6	16CSE45	Computer Organization	3	0	0	1	4	3	0	50	50	100
7	16CSE46	Mini Project	0	2	0	0	2	0	4	50	50	100
Total							**24**	**17**	**12**	**375**	**375**	**750**

THIRD SEMESTER

(SYLLABUS)

ENGINEERING MATHEMATICS – III
(Common to All Branches)

Course Code	**: 16MAT31**	**Credits**	**05**
L:P:T:S	**: 4:0:1:0**	**CIE Marks**	**50**
Exam Hours	**: 3**	**SEE Marks**	**50**

Course Outcomes: At the end of the Course, the student will be able:

CO1	Solve the Fourier series expansion of functions analytically and numerically.
CO2	Solve the Continuous model problems using Fourier transform.
CO3	Solve the discrete model problems using Z-transforms and Fast Fourier transform.
CO4	Fit a suitable curve by the method of least squares and determine the lines of regression for a set of statistical data.
CO5	Use appropriate numerical methods to solve algebraic and transcendental equations and also to calculate a definite integral numerically.
CO6	Use appropriate numerical methods to solve Boundary Value Problems in Partial differential equations.

Mapping of Course Outcomes to Program Outcomes:

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**POI0**	**POII**	**PO12**
CO1	3	3	1	1	1	-	-	-	2	1	-	2
CO2	3	3	1	1	1	-	-	-	1	1	-	2
CO3	3	3	2	3	3	-	-	-	2	1	-	2
CO4	2	3	2	2	2	-	-	-	1	3	-	1
CO5	2	2	3	3	2	-	-	-	1	2	-	1
CO6	3	3	3	2	3	-	-	-	2	1	-	1

Module No	Module Contents	Hours	COs
1	**Fourier series**: Periodic function, Dirichlet's conditions, Fourier series of periodic functions of period 2 and arbitrary period $2l$, half range series. Fourier series and half Range Fourier series of periodic square wave, half wave rectifier, full wave rectifier, Saw-tooth wave with graphical representation, practical harmonic analysis.	9	CO1
2	**Fourier Transforms:** Infinite Fourier transforms, Fourier Sine and Cosine transforms, Inverse Fourier transform. **Z - Transform:** Definition, Z-transforms of some standard functions, properties, damping rule, shifting rule (without proof), initial and final value theorems, inverse Z- transforms. **Applications:** Solving difference equations using Z-transform.	9	CO2, CO3
3	**Statistical Methods:** Fitting of the curves of the form $y = a + bx$, $y = a + bx + cx^2$, $y = ae^{bx}$, $y = ax^b$, and $y = ab^x$ by the method of least square**,** Correlation and Regression , Regression coefficients, line of regression – problems. **Discrete Fourier Transform and Fast Fourier Transform**: Definition of N-Point DFT, problems for 4-Points and inverse DFT for four points only. FFT algorithm to compute the Fourier transforms 4-Point only.	9	CO3, CO4
4	**Numerical Methods-1:** Numerical solution of algebraic and transcendental equations; Rugula- falsi method and Newton Raphson's method. Solution of a system of equations using Gauss-seidel and Relaxation method. Interpolation & extrapolation – Newton's forward and backward formulae for equal intervals, Newton divided difference and Lagrange's formulae for unequal	9	CO5

	intervals.		
5	**Numerical Methods-2**: Numerical integration - Simpson's 1/3rd rule, Simpson's 3/8th rule, Weddle's rule (without proof)-Problems. Numerical solution of Boundary value problems-Solution of one dimensional wave equation and heat equation, Numerical solution of two dimensional Laplace's equation and Poisson's equation.	9	CO5 CO6

TEXT BOOKS:

1. Advanced Engineering Mathematics, Erwin Kreyszig, Wiley-India publishers, 10th edition, 2014.
2. Higher Engineering Mathematics, B.S.Grewal, Khanna Publishers, 43rd edition, 2014.

REFERENCE BOOKS:

1. Advanced Modern Engineering Mathematics, Glyn James, Pearson Education, 4th edition, 2015.
2. Advanced Engineering Mathematics, Dennis G. Zill, Michael R. Cullen, Jones and Barlett Publishers Inc., 4th edition, 2015,.
3. Engineering Mathematics, B.V.Ramana, Tata McGraw Hill Publications, 4th edition, 2005.
4. Engineering Mathematics, Anthony Craft, Pearson Education, 4th edition, 2013.

Assessment Pattern

CIE- Continuous Internal Evaluation (50 Marks)

Bloom's Taxonomy	**Tests**	**Assignments**	**Quizzes**
Marks (out of 50)	30	10	10
Remember	10	3	5
Understand	5	5	5
Apply	5	2	-
Analyze	5	-	-
Evaluate	5	-	-
Create	-	-	-

SEE – Semester End Examination (50 Marks)

Bloom's Taxonomy Marks (Out of 50)	Tests
Remember	10
Understand	10
Apply	20
Analyze	5
Evaluate	5
Create	-

ANALOG AND DIGITAL ELECTRONICS

Course Code : 16CSE33 **Credits : 05**
L:P:T:S : 3:2:0:0 **CIE Marks : 50+25**
Exam Hours : 3+3 **SEE Marks : 50+25**

Course Outcomes: At the end of the Course, the Student will be able to:

CO1	alyze the operation and understand the working of various electronic components and electronic circuits.
CO2	plement Boolean function using Karnaugh maps and Quine Mc-Clusky method.
CO3	sign and Analyze modular combinatorial logic circuits.
CO4	velop Bi- stable elements like flip-flop and use its functionality to understand the sequential circuits and its applications.
CO5	sign and apply the concepts of state and state transition for the analysis of sequential circuits.
CO6	nstruct Verilog code to implement the combinational and sequential circuits.

Mapping of Course Outcomes to Program Outcomes and Program Specific Outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	-	-	3	-	-	-	-	-	-	-	-	-
CO2	3	-	3	3	-	-	-	-	-	-	-	-	3	-
CO3	3	3	3	3	-	-	-	-	-	-	-	-	3	-
CO4	3	3	3	3	-	-	-	-	-	-	-	-	3	-
CO5	3	3	3	3	-	-	-	-	-	-	-	-	3	-
CO6	3	3	3	3	3	-	-	-	-	-	-	1	3	-
Average	3	3	3	3	3	-	-	-	-	-	-	1	3	-

Sl. No.	Contents of Module	Hrs	COs
1	**Electronic devices and applications** BJT vs FET, MOSFETs, CMOS Device, Ideal vs practical OP-AMP, Comparator, Active Filters, Relaxation Oscillator Optoelectronics devices: Photo-conductors, Photo-diodes, LED, LCD, CRT	9	CO1
	List of experiments 1. Design and implement Schmitt trigger for given UTP and LTP. Also, implement using a simulation package. 2. Design and construct Op- Amp relaxation oscillator for given frequency and demonstrate its working. Also, implement using a simulation package.	8	
2	**Wave Shaping Circuit** RC as Low pass and High pass, RC as Integrator and Differentiator, Diode as Clipper and Clamper, Bistable Multivibrators, IC Multivibrators: Astable and Monostable, Clock Waveforms	9	CO1
	List of experiments 1. Design and implement clipper and clamper (positive and negative for both) use diodes. Also, demonstrate the working using simulation package. **2.** Design and implement an astable multivibrator circuit using 555-timer for a given frequency and duty cycle. Also, implement using simulation package.	9	
3	**Combinational Logic Circuits** Karnaugh maps, Quine-McClusky method, Half-adder, Full adder, Subtractor, Multiplexers and De-multiplexers, Magnitude comparators (1 and 2 bit), Design of multiple output circuits using PLDs. Introduction to HDL, Verilog Implementation of Data Processing Circuits	9	
	List of experiments 1. Given a 4-variable logic expression, simplify it using Entered Variable Map and realize the simplified logic expression using 8:1 multiplexer IC. Simulate and verify its working using Verilog code. 2. Perform n bit addition / subtraction using 4-bit full adder IC. Simulate and verify its working using Verilog code.	9	CO2, CO3, CO6
4	**Sequential Circuit Elements** Latches, types of Flip-flops, Flip-flop excitation tables, Registers, type of Shift Registers, Universal shift Registers, Applications of Shift Registers –Ring Counter, Johnson Counter, Sequence generator, Verilog implementation of Flip-flops and Registers.	9	
	List of experiments 1. Realize JK, D and T Flip-Flops and verify its truth table Simulate and verify the working of the same using VERILOG code.	9	CO4, CO6

	2. Design and implement Ring counter and Johnson counter using 4-bit shift register and demonstrate its working. Simulate and verify the working using VERILOG code.		
5	**Analysis of Sequential Circuit** Counters-Asynchronous and synchronous, Design of counters, Counter Design as synthesis problem, Design of Synchronous Sequential Circuits: Moore model, Mealy model, State Reduction Techniques, Verilog implementation of counters.	**9**	**CO4, CO5, CO6**
	List of experiments 1. Design and implement a mod-n (n<8) synchronous up or down counter using J-K Flip-Flop ICs and demonstrate its working. Simulate and verify mod 8 synchronous up or down counter using VERILOG code. 2. Design and implement an asynchronous counter using decade counter IC to count up from 0 to n (n<=9) and demonstrate its working.	**9**	

Text Books

1. Electronic Devices and Circuits, Anil K Maini, Varsha Agarwal, 1st Edition, 2009, Wiley.
2. Digital Principles and Applications, Donald P Leach and Albert Paul Malvino, 8th Edition, 2014, Tata McGraw Hill.

Reference books:

1. Digital Principles and design, Donald D. Givone, 2003, Tata McGraw Hill.
2. Digital Design: with an Introduction to Verilog HDL, M Morris Mano and Michael D. Ciletti, 5th Edition, 2013, Pearson Education.
3. Integrated Electronics – Analog and Digital Circuits and Systems, Jacob Millman, Christos Halkias and Chetan D Parikh, 2nd Edition, 2011, Tata McGraw Hill.

CIE- Continuous Internal Evaluation: Theory (50 Marks)

Blooms Taxonomy	Tests	Assignments	Quizzes
Marks (Out of 50)	**30**	**10**	**10**
L1: Remember	**--**	**--**	**--**
L2: Understand	**10**	**5**	**--**
L3: Apply	**15**	**5**	**5**
L4: Analyse	**5**	**--**	**5**
L5: Evaluate	**--**	**--**	**--**
L6: Create	**--**	**--**	**--**

CIE- Continuous Internal Evaluation: Lab (25 Marks)

Blooms Taxonomy	Marks (Out of 25)
L1: Remember	--
L2: Understand	05
L3: Apply	05
L4: Analyse	10
L5: Evaluate	05
L6: Create	--

SEE- Semester End Examination: Theory (50Marks)

Blooms Taxonomy	Marks (Out of 50)
L1: Remember	05
L2: Understand	05
L3: Apply	20
L4: Analyze	10
L5: Evaluate	05
L6: Create	05

SEE- Semester End Examination: Lab (25Marks)

Blooms Taxonomy	Marks (Out of 25)
L1: Remember	--
L2: Understand	--
L3: Apply	10
L4: Analyze	10
L5: Evaluate	05
L6: Create	--

DATA STRUCTURES USING C

Course Code : 16CSE34 **Credits: 05**
L:P:T:S : 3:2:0:0 **CIE Marks: 50+25**
Exam Hours : 3+3 **SEE Marks: 50+25**

Course Outcomes: At the end of the Course, the Student will be able to:

CO #	COURSE OUTCOME
CO1	Analyze the concept of array data structures, its applications and dynamic memory management.
CO2	Compare and analyze different sorting techniques and apply them in organizing the data.
CO3	Analyze the concepts of stacks, queues and linked list in problem solving.
CO4	Design applications using non-linear data structures.
CO5	Analyze various advanced tree and graph concepts.
CO6	Develop algorithms to solve problems using fundamental data structures.

Mapping of Course Outcomes to Program Outcomes and Program Specific Outcomes:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	-	-	-	-	-	-	-	-	-	3	3	3
CO2	3	3	-	-	3	-	-	-	3	-	-	3	3	3
CO3	3	3	-	-	3	-	-	-	3	-	-	-	3	3
CO4	3		3	3	3	-	-	-	3	-	-	-	3	3
CO5	3	3	-	-	3	-	-	-	3	-	-	-	3	3
CO6	3	3	3	3	-	-	-	-	-	-	-	3	3	3

Correlation levels: 1-Slight (Low) 2-Moderate (Medium) 3-Substantial (High)

Module No	Module Contents	Hours	COs
1	**Introduction to Data Structures:** Arrays and Pointers revisited, Sparse matrix, transpose of a sparse matrix, dynamic memory management. Introduction to Data Structures, Classification of Data Structures, Abstract Data Types, Insertion sort, Quick sort, Shell sort, Radix sort.	9	CO1, CO2, CO6
	List of programs: 1. Write a program to check whether matrix is sparse or not. 2. Write a program to determine the transpose of a sparse matrix.	8	

2	**Stacks & Queues:** Stacks: Definition, Stack representation, Primitive operations on stack, array representation of stacks Applications of stacks: Recursion, Fibonacci series, Tower of Hanoi problem, Conversion of expressions, Evaluation of postfix expression, Iteration v/s recursion Queues: Definition, Queue representation, Primitive operations on queue, array representation of queues, Circular queue, Priority queue, Double ended queue, Applications of queues.	9	CO3, CO6
	List of programs: 1. Write a program to demonstrate Tower of Hanoi problem 2. Write a program for Ackermann's function 3. Develop a program for STACK that performs following primitive operations: push, pop and display 4. Develop a program to convert INFIX notation to POSTFIX 5. Develop a program for evaluation of POSTFIX notation. 6. Develop a program for QUEUE that performs following primitive operations: insert, delete and display 7. Develop a program for CIRCULAR QUEUE that performs following primitive operations: insert, delete and display	9	
3	**Linked Lists:** Dynamic memory allocation revisited – malloc, calloc, realloc, free, Introduction to linked list, Representation of linked list in memory, primitive operations on linked list, searching a linked list, circular linked list, doubly linked list, header linked list Applications of linked list: Josephus problem, addition of two long integers, addition of two polynomials, Linked representation of stack, Linked representation of queue.	9	CO1, CO3, CO6
	List of programs: 1. Write a menu driven program to perform primitive operations on single linked list 2. Write a program to reverse a single linked list 3. Develop a program for addition of two long integers using linked list.	9	
4	**Trees-I:** Introduction, Binary tree – strictly binary tree, complete binary tree, representing binary tree in memory, traversing a binary tree, binary search tree, insertion and deletion in binary search tree, threaded binary tree. Expression trees, construction of an expression tree from prefix and postfix, Heap tree, creation of heap tree, insertion in heap, Deletion from heap.	9	CO4, CO6

	List of programs: 1. Develop a program to traverse a tree using in-order, pre-order and post-order. 2. Develop a program to perform insertion, deletion and traversal of a binary search tree	9	
5	**Trees-II & Graphs:** AVL Trees, Rotations in AVL tree, Insertion and deletion in an AVL tree, Huffman's algorithm. Introduction to Graph, Graph theory terminologies, sequential representation of a graph, adjacency matrix and path matrix, Warshall's algorithm, Linked representation of a graph, Operations on a graph, Traversing a graph, Topological sorting	9	CO4, CO5, CO6
	List of programs: 1. Develop a program to implement BFS traversal of graph 2. Develop a program to implement DFS traversal of graph	9	

TEXT BOOKS:

1. Data Structures with C, Seymour Lipschutz, McGraw Hill Education, Special Indian Edition, Thirteenth reprint 2015.
2. Data Structures using C, Aaron M. Tanenbaum, Yedidyah Langsam & Moshe J Augenstein, Pearson Education, Thirteenth Impression 2014.

REFERENCE BOOKS:

1. Data Structures – A Pseudocode Approach with C, Richard F Gilberg and Behrouz A Forouzan, Cengage Learning , Second edition, Fifth Indian Reprint 2015,

CIE - Continuous Internal Evaluation: Theory (50 Marks)

Bloom's Taxonomy	Tests	Assignments	Quizzes
Marks (Out of 50)	30	10	10
Remember	5	-	5
Understand	10	5	-
Apply	15	5	5
Analyze	-	-	-
Evaluate	-	-	-
Create	-	-	-

CIE - Continuous Internal Evaluation: Lab (25Marks)

Bloom's Taxonomy	Lab
Marks (Out of 25)	25
Remember	-
Understand	-
Apply	15-
Analyze	-10
Evaluate	-
Create	

SEE –Semester End Examination: Theory (50 Marks)

Bloom's Taxonomy	Tests
Marks (Out of 50)	
Remember	5
Understand	20
Apply	25
Analyze	-
Evaluate	-
Create	-

SEE – Semester End Examination: Lab (25 Marks)

Bloom's Taxonomy	Lab
Marks (Out of 25)	
Remember	-
Understand	-
Apply	15
Analyze	10
Evaluate	-
Create	-

UNIX & SHELL PROGRAMMING

Course Code: 16CSE35 **Credits : 4**
L:P:T:S: 3:1:0:0 **CIE Marks : 50+25**
Exam Hours: 3 **SEE Marks : 50+25**

Course Outcomes: At the end of the Course, the Student will be able to:

CO #	COURSE OUTCOME
CO1	Apply the basic utility commands of UNIX operating system.
CO2	Use &analyze the various VI editor commands.
CO3	Analyze the process creation mechanism in Unix and create the various file/directory handling commands.
CO4	Create various filter commands and regular expression which can be used for quick data retrieval for various societal applications.
CO5	Design and develop shell scripts for the society with good code of ethics.
CO6	Formulate awk commands for the benefit of society.

Mapping of Course Outcomes to Program Outcomes:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	-	-	-	-	-	-	-	-	-	-	1	-	1
CO2	3	3	-	-	-	-	-	-	-	-	-	-	3	-
CO3	3	3	3	-	-	-	-	-	-	-	-	-	3	-
CO4	3	3	3	-	-	-	-	-	2	-	-	-	3	-
CO5	3	3	3	-	-	-	-	1	2	-	-	-	3	-
CO6	3	3	3	-	-	-	-	-	2	-	-	-	3	-

Module	Contents of Module	Hours	COs
1	**Getting started & Understanding UNIX commands:** Operating System, UNIX Operating System, UNIX architecture, features of UNIX, Knowing your machine & keyboard, System Administrator, Logging in & out, working out with commands, understanding UNIX commands.	9	CO1
	List of programs: 1. Knowing your UNIX (FEDORA / UBUNTU) system 2. Create your user name (account) and password. 3. Login to your account and go through the environment.	8	
2	**General Purpose Utilities:** passwd, who, tty, lock, sty, script, clear antput, uname, date, cal, calendar, bc VI editor: preliminaries, quitting VI, inserting an replacing text, saving text, exit to UNIX shell, repeat factor, command mode, navigation, operators, deleting, moving and yanking text, changing text, repeating last command, undoing last editing instructions, string search, search with regular expressions, search and replace	9	CO2
	List of programs: 1. Execution of various general purpose utility commands as studied in Module 2. 2. Hands on session on VI editor – modes of functioning, replace text characters, saving your work, quitting, navigation across the file, use of operator-command combinations to copy, move and delete text, undo the last command, search and replace patterns	9	
3	**File system, Attributes& Process:** File system: The file, parent child relationship, UNIX file system, pwd, absolute pathname, changing directories, relative pathname, mkdir, rmdir, cp, rm, mv, cat, file, lp, cancel, df, du, compress, gzip, zip File Attributes: ls, ls –l, file permissions, chmod, directory permissions, umask, file ownership, chown and chgrp, file modification and access times, touch, ln, symbolic links, find Understanding the Process, how process is created, login shell, init process, internal and external commands, process status, running jobs in background, nice, signals, kill, at and batch, cron	9	CO2

	List of programs: 1. Execution of various file/directory handling commands as studied in Module 3.	9	
4	**Simple filters & Regular expressions:** more, wc, od, pr, cmp, diff, comm, head, tail, cut, paste, sort, tr, uniq, nl grep – searching for a pattern, grep options, regular expressions, egrep and fgrep sed – stream editor, line addressing, context addressing, editing text, substitution, regular expressions.	9	CO3, CO4
	List of programs: 1. Execution of various filter commands as studied in Module 4.	9	
5	**Shell Programming& AWK:** Shell variables, shell scripts, read, positional parameters, exit status, logical operators, exit, if conditions, test and [], case, expr, sleep and wait, while and for AWK preliminaries, splitting line into fields, printf – formatting output, comparison operators, number processing, variables, reading program from a file, BEGIN and END section, positional parameters, getline, built in variables, arrays, functions, control flow, looping	9	CO5, CO6
	List of programs: 1. Write a shell script to check how many users are currently working in UNIX system. 2. Write a shell script to check if current directory is same as your HOME directory. 3. Write a shell script to accept a file and check if it is executable. If not make it executable 4. Write a shell script which will accept a filename and starting and ending line numbers and displays these lines from given file. 5. Write a shell script which displays a list of all the files in the current directory to which you have read, write and execute permissions. 6. A shell script receives even number of filenames as arguments. Suppose four files are supplied as arguments then the first file should get copied into second, third file into fourth and so on. If odd number of filenames is supplied then no copying should take place and an error message should be displayed. 7. Write a shell script which will receive any number of filenames as arguments. The shell script should check whether every argument supplied is a file or a directory. If it a directory it should be appropriately reported. If it is a filename then name of the file as well as the number of lines present in it should be reported. 8. Write a shell script which gets executed the moment the user logs in. It should display the message, " Good Morning", " Good Afternoon", " Good Evening", depending upon the time at which the user logs in.	9	

	9. Write a shell script which will receive login name during execution, obtain information about it from /etc/passwd and display this information on screen in easily understandable format. 10. Write a shell script which accepts any number of arguments and prints them in reverse order. Ex: If file name is test then $sh test A B C should produce C B A. 11. Write scripts to demonstrate built in variables available in AWK 12. Write scripts to demonstrate built in functions available in AWK 13. Create any one of the database file (library, student information, cricket information etc) of your choice and answer the queries asked. 14. NOTE: The database and queries will be given during respective lab hours.		

TEXT BOOKS:

1. Your UNIX – The ultimate Guide, SUMITABHA DAS, McGraw Hill, TATA McGraw Hill, 3rd Edition, reprint 2012.
2. UNIX – Concepts & Applications, SUMITABHA DAS, McGraw Hill, TATA McGraw Hill Edition, Fourth edition, reprint 2

REFERENCE BOOKS:

1. UNIX and SHELL Programming, Richard F Gilberg and Behrouz A Forouzan, Cengage Learning, 15th impression, 2015.

CIE - Continuous Internal Evaluation: Theory (50 marks)

Bloom's Taxonomy	Tests	Assignments	Quizzes
Marks (Out of 50)	**30**	**10**	**10**
Remember	5	-	5
Understand	5	5	5
Apply	5	-	-
Analyze	10	5	-
Evaluate	5	-	-
Create	-	-	--

CIE - Continuous Internal Evaluation: Lab (25marks)

Bloom's Taxonomy	Lab
Marks (Out of 25)	
Remember	-
Understand	-
Apply	15
Analyze	-
Evaluate	10
Create	-

SEE –Semester End Examination: Theory (50 Marks)

Bloom's Taxonomy	Tests
Remember	5
Understand	30
Apply	5
Analyze	10
Evaluate	5
Create	-

SEE – Semester End Examination: Lab (25 Marks)

Bloom's Taxonomy	Lab
Marks (Out of 25)	
Remember	-
Understand	-
Apply	15
Analyze	-
Evaluate	10
Create	-

16CSE36 - MINI PROJECT

Course Code : 16CSE36 **Credits : 02**
L: P: T: S : 0:2:0:0 **CIE Marks : 25**
Exam Hours : 3 **SEE Marks : 25**

Course Outcomes: At the end of the Course, the Student will be able to

CO #	COURSE OUTCOME
CO1	Apply the knowledge on the operations of various data structures.
CO2	Compare and contrast different sorting techniques and its applications.
CO3	Write and analyze algorithms for the problem statement.
CO4	Implement operations like searching, insertion, and deletion, traversing mechanism etc. on one or more data structures.
CO5	Create a software solution for real time application using one or more data structures.
CO6	Demonstrate their communication skill effectively with technical presentation.

Mapping of Course Outcomes to Program Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	-	-	-	3	-	-	3	3	3	-	-	-	-
CO2	3	3	3	3	3	-	-	3	3	3	-	-	3	3

CO3	3	3	3	-	3	-	-	3	3	3	-	3	3	3
CO4	3	3	3	3	3	-	-	3	3	3	-	3	3	3
CO5	3	3	3	3	3	-	-	3	3	3	-	3	3	3
CO6	3	-	-	-	-	-	-	3	3	3	-	-	-	-

The student shall be capable of identifying a problem related to the field of Computer Science and carry out a mini project on the problem defined. Each student is expected to do the mini project individually. The code developed towards the project will be reviewed by a panel of experts during the course of the semester. Plagiarized projects will automatically get an **"F" GRADE** and the student will be liable for further disciplinary action. At the completion of a project the student will submit a project report, which will be evaluated by duly appointed examiner(s).

Sample Mini project includes:

1. Tic-Tac-Toe Game
2. Quiz Game
3. Library Management
4. Telecom Billing Management system
5. Numerical Method Applications

CIE - Continuous Internal Evaluation (25 Marks) (50Marks)

Bloom's Taxonomy	**Mini Project**
Marks (Out of 25)	**-**
Remember	-
Understand	6
Apply	14
Analyze	10
Evaluate	10
Create	10

SEE – Semester End Examination (25 Marks) (50Marks)

Bloom's Taxonomy	**Mini Project**
Remember	-
Understand	6
Apply	14
Analyze	10
Evaluate	10
Create	10

FOURTH SEMESTER

(SYLLABUS)

ENGINEERING MATHEMATICS – IV
(Common to All Branches)

Course Code: 16MAT41 **Credits: 05**
L: P: T: S : 4:0:1:0 **CIE Marks:50**
Exam Hours: 03 **SEE Marks: 50**

Course Outcomes: At the end of the Course, the Student will be able to do the following:

CO1	Solve initial value problems using appropriate numerical methods.
CO2	Understand the concepts of Complex variables and transformation for solving Engineering Problems.
CO3	Understand the concepts of complex integration, Poles and Residuals in the Stability analysis of engineering problems.
CO4	Gain ability to use probability distributions to analyze and solve real time problems.
CO5	Apply the stochastic process and Markov Chain in prediction of future events.
CO6	Analyze, interpret, and evaluate scientific hypotheses and theories using rigorous probability and statistical methods.

Mapping of Course Outcomes to Program Outcomes:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	3	2	2	-	-	-	2	1	-	2
CO2	3	3	3	2	2	-	-	-	1	1	-	1
CO3	3	3	2	3	2	-	-	-	2	1	-	2
CO4	3	2	2	2	3	-	-	-	1	3	-	1
CO5	2	2	3	3	2	-	-	-	1	3	-	1
CO6	3	3	3	2	3	-	-	-	2	2	-	1

Module No.	Module Contents	Hours	COs
1	**Numerical Methods:** Numerical solution of ordinary differential equations of first order and of first degree: single step methods- Picard's Method, Taylor's series method, modified Euler's metho and Runge-Kutta method of fourth-order. Multi step methods- Milne's and Adams- Bashforth predictor and corrector methods. Numerical solution of simultaneous first order differential	9	CO1

	equations ; Picard's Method and Runge-Kutta Method of fourth order(no derivation of formulae)		
2	**Complex Variables**: Functions of complex Variables, Analytical functions, Cauchy's Riemann Equations in Cartesian and Polar forms, Harmonic functions and Construction of analytic function Discussion of Transformations: $w = z^2$, $w = e^z$ and $w = z + (1 / z)$ and Bilinear Transformations.	9	CO2
3	**Complex Integrations:** Complex line integrals – Cauchy's theorem and Cauchy's Integral formula. Power Series, Laurent's series. Singularities, Poles and Residuals, Residual Theorem-problems (Without proof).	9	CO3
4	**Probability distributions:** Random variables (discrete and continuous), probability density function, cumulative density function. Discrete Probability distributions: Binomial and Poisson distributions. Continuous Probability distributions; Exponential and normal distributions. **Joint Probability distributions:**, Mathematical expectation, correlation, covariance (discrete random variables only).	9	CO4
5	**Sampling Theory:** Sampling, Sampling distributions, standard error, test of hypothesis for means and proportions, confidence limits for means, student's t-distribution, Chi-square distribution for test of goodness of fit. **Stochastic Processes:** Stochastic processes, Probability Vectors, Stochastic matrix, Regular stochastic matrix, Markov chains, Higher transition probabilities, Stationary distribution of regular Markov chains and absorbing states	9	CO5, CO6

TEXT BOOKS:

1. Advanced Engineering Mathematics, Erwin Kreyszig, Wiley-India publishers, 10th edition, 2014.
2. Higher Engineering Mathematics, B.S.Grewal, Khanna Publishers, 43nd edition, 2014.

REFERENCE BOOKS:

1. Advanced Modern Engineering Mathematics, Glyn James, Pearson Education, 4th edition, 2015.
2. Advanced Engineering Mathematics, Dennis G. Zill, Michael R. Cullen, Jones and Barlett Publishers Inc, 4th edition, 2015,
3. Engineering Mathematics, B. V. Ramana, Tata McGraw Hill Publications, 4th edition, 2005.
4. Engineering Mathematics, Anthony Craft, Pearson Education, 4th edition, 2013.

Assessment Pattern

CIE- Continuous Internal Evaluation (50 Marks)

Bloom's Category / Marks (Out of 50)	Tests (30 Marks)	Assignments (10 Marks)	Quizzes (10 Marks)
Remember	10	3	5
Understand	5	5	5
Apply	5	2	-
Analyze	5	-	-
Evaluate	5	-	-
Create	-	-	-

SEE- Semester End Examination (50 Marks)

Bloom's Category / Marks (Out of 50)	Test (50 Marks)
Remember	10
Understand	10
Apply	20
Analyze	5
Evaluate	5
Create	-

OBJECT ORIENTED PROGRAMMING WITH C++

Course Code : 16CSE43 **Credits : 05**
L: P: T: S : 3:2:0:0 **CIE Marks : 50+25**
Exam Hours : 3+3 **SEE Marks : 50+25**

COURSE OUTCOMES: At the end of the Course, the Student will be able to

CO #	COURSE OUTCOMES
CO1	Identify potential benefits of object-oriented programming over structured programming.
CO2	Apply concepts of classes and objects to correlate their significance in real world.
CO3	Construct classes using function overloading and operator overloading.
CO4	Design programs using inheritance and run time polymorphism.
CO5	Apply virtual and pure virtual functions to solve complex problems.
CO6	Analyze benefits of generic classes with C++ templates.

Course Outcomes to Program Outcomes Articulation Matrix

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	-	-	-		-	-	-		-	-	-	-	-
CO2	3	3	-	-	3	-	-	-	3	-	-	1	3	-
CO3	3	3	2	-	3	-	-	-	3	-	-	-	3	-
CO4	3	3	2	1	3	-	-	-	3	-	-	-	3	-
CO5	3	3	2	1	3	-	-	-	3	-	-	-	3	-
CO6	3	3	-	-	3	-	-	-	3	-	-	-	3	-

Module No.	Module Contents	Hours	COs
1	**Introduction to Object-Oriented Programming:** Evolution of programming methodologies, Procedural Approach V/s Object-Oriented Approach. Principles of OOP: Encapsulation and Abstraction, Message Passing, Inheritance– Reusability, Extensibility, Polymorphism – Overloading, Dynamic Binding **Comparison of C and C++:** Limitations of C, Introduction to C++, Structure of the C++ program, Added features of C++ over C – Storage Classes, Reference variables, Inline functions. Data types – control structures – Arrays and Strings – User defined types – Functions and Pointers. **Review of Basic Language Constructs**: Data types – control structures– Arrays and Strings – User defined types – Functions and Pointers.	9	CO1
	List of Experiments **1.** Write a C++ program to find the largest of three numbers using inline function. **2.** Write a C++ program to sort an array of integer in ascending order using a function called exchange() which accepts two integer arguments by reference. **3.** Write a C++ program to demonstrate the static and non-static variable usage defining them within a function.	8	
2	**Introduction to Objects and Classes** Defining the class, Defining Data members and member functions, Creating Objects of Class, Access Specifiers. Scope Resolution Operator, Friend Functions and Friend Classes, Static Members, this pointer, returning values using this pointer. Comparison of class with structure. **Constructors and Destructors** Purpose of Constructors and	9	CO2

	Destructors, Default Constructors, Constructors with & without parameters, Constructor Overloading, Copy Constructor. Invoking Constructors and Destructors. **Pointers in C++** Pointer declaration and Access, pointer and arrays, pointer to functions, memory management – new and delete.		
	List of Experiments **1.** Design, develop, and execute a program in C++ based on the following requirements: An EMPLOYEE class is to contain the following data members and member functions: Data members: Employee_Number (an integer), Employee_Name (astring of characters), Basic_Salary (an integer),All_Allowances (an integer), IT (an integer), Net_Salary (an integer). Member functions: to read the data of an employee, to calculate Net_Salary and to print the values of all the data members. (All_Allowances = 123% of Basic; IncomeTax (IT)= 30% of the gross salary (=basic_Salary _All_Allowance); Net_Salary = Basic_Salary + All_Allowances – IT) **2.** Write a C++ program to perform matrix addition using static variable, default argument and friend function. **3.**Write a C++ program for matrix manipulation with dynamic memory allocation using copy constructor and overloading of assignment operator **4.** Create a class 'COMPLEX' to hold a complex number. Write a friend function to add two complex numbers. Write a main function to add two COMPLEX objects.	9	
3	**Polymorphism** Overloading Concepts Function Overloading: Functions with different sets of parameters, default and constant parameters. Operator Overloading: Rules for overloading Operators. Overloading unary operators, overloading binary operators, Overloading Comma, [], (), ->, new, delete Operators. Type Conversions.	9	CO3
	List of related Experiments **1.** Write a C++ program to implement function overloading in order to compute power(m, n) where i) m is double and n is int ii) m and n are int. **2.** Create a 'STRING' class which overloads ' = = ' operator to compare two STRING objects **3.**Write a C++ program to overload new and delete operators.	9	
4	**Inheritance** Basic Concepts, Reusability & Extensibility. Defining derived classes, protected access specifier in Base class – public, private & protected inheritance – constructors and destructors in derived	9	CO4, CO5

	classes – Types of Inheritances. Virtual base class. **Virtual Functions** Normal member functions accessed with pointers, virtual member function access, late binding, pure virtual function, abstract classes.		
	List of Experiments **1.** Create a 'MATRIX' class of size m X n. Overload the '+' operator to add two MATRIX objects **2.** Derive a class 'MAT' from MATRIX class created in the above program. Add a member function to overload '*' operator to multiply two objects(Single Inheritance).	9	
5	**Templates** Generic Functions- A generic swap function, Functions with more than one Generic Type, Overloading a Function Template. Generic Classes – A stack generic class, Class template with more than one Generic Type, type name and template keywords, Template Restrictions, The power of Templates. **Exception Handling-** Fundamentals of Exception Handling, Catching Class Types, Using Multiple catch statements, Catching All Exception, Restricting Exception, throw statement, Setting the Terminate and Unexpected Handlers, Uncaught exception, bad exception Classes, and Built-In Exceptions. Exception Vs Error Handling, Assertion in C++.	9	CO6
	List of Experiments **1.** Write a C++ program for bubble sort using template. **2.** Define a function template for finding the minimum value contained in an array. Write main( ) function to find the minimum value of integer array and minimum value of floating point numbers in an array.	9	

TEXT BOOKS:

1. C++ How to Program, Paul Deitel, Harvey Deitel, Pearson Education Limited ,9th Edition, 2016
2. Object Oriented Programming with C++, E Balagurusamy, ,TMH, 6th Edition, 2013

REFERENCE BOOKS:

1. C++ Primer Plus, Stephen Prata, Pearson Education Limited, 6th Edition, 2015.
2. C++ PROGRAMMING Today, Barbara Johnston, Pearson Education, 2nd Edition, 2015.

CIE- Continuous Internal Evaluation: Theory (50 Marks)

Bloom's Category	Tests	Assignments	Quizzes
Marks (out of 50)	30	10	10
Remember	5	-	5
Understand	5	5	-

Apply	10	-	5
Analyze	5	5	-
Evaluate	-	-	-
Create	5	-	-

CIE- Continuous Internal Evaluation: Lab (25 Marks)

Bloom's Category	**Lab**
Marks (out of 25)	
Remember	-
Understand	-
Apply	-
Analyze	-
Evaluate	25
Create	-

SEE- Semester End Examination: Theory (50 Marks)

Bloom's Category	**Theory**
Marks (out of 50)	
Remember	10
Understand	10
Apply	20
Analyze	5
Evaluate	-
Create	5

SEE- Semester End Examination: Lab (25 Marks)

Bloom's Category	**Lab**
Marks (out of 25)	
Remember	-
Understand	-
Apply	15
Analyze	-
Evaluate	10
Create	-

MICROPROCESSOR

Course Code : 16CSE44 **Credits : 05**
L:P:T:S : 3:2:0:0 **CIE Marks : 50+25**
Exam Hours : 3+3 **SEE Marks : 50+25**

COURSE OUTCOMES: At the end of the Course, the Student will be able to:

CO #	COURSE OUTCOME
CO1	Summarize the architecture and organization of 8086.
CO2	Apply the various instructions of 8086.
CO3	Analyze instruction set and develop assembly level programs of 8086 and coprocessor 8087.
CO4	Analyze operation modes, memory and I/O address decoding concepts of 8086.
CO5	Design software interface to control hardware devices.
CO6	Compare and contrast advanced microprocessors.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	-	-	-	-	-	-	-	-	-	-	3	3	-
CO2	3	3	3	3	-	-	-	-	-	-	-	3	3	-
CO3	3	3	3	3	-	-	-	-	-	3	-	3	3	3
CO4	3	3	-	-	-	-	-	-	-	-	-	3	3	-
CO5	3	3	3	3	-	-	-	-	-	3	-	3	3	3
CO6	3	3	-	-	-	-	-	-	-	-	-	-	3	-

Module No.	Module Contents	Hours	COs
1	**8086 architecture**: Evolution of Microprocessor, Internal architecture of 8086; Programming Model, Real mode memory addressing, Addressing modes, Data transfer instructions, Arithmetic instructions	9	CO1
	List of experiments 1) Program to perform 32 bit addition, multiplication and division operations. 2) Program to sort a given array of N elements is ascending / descending order using bubble and insertion sort 3) Program to check a given string is palindrome or not 4) Program to implement digital clock.	8	

2	**8086 Instruction set**: Assembler directives, Logical instructions, Program control instructions, String Manipulation instructions, Miscellaneous instructions, and programs, Introduction to interrupts	9	CO2, CO3
	List of experiments 1) Program to validate an entered string. 2) Program to perform concatenation of two given strings. 3) Program to generate Fibonacci series of N numbers. 4) Program to compute factorial and $^{n}C_{r}$ using recursion.	9	
3	**Hardware Specification**: Pin Configuration, Clock Generator, Timing diagrams, 8288 Bus, Interrupts, Physical memory, Types of memory, Memory decoding	9	CO4
	List of experiments 1) Program to read and set system time. 2) Program to read current date and set the date. 3) Program to perform file operations such as open, close, create, delete and rename a file. 4) Program to link modules present in different files.	9	
4	**Interfacing** : Basic I/O interfacing, I/O address decoding, 8255 programmable peripheral interface, bus protocols like PCI, serial com, USB	9	CO4, CO5
	List of experiments 1) Program to implement addition and multiplication on logic controller interface. 2) Program to rotate stepper motor clockwise N times and rotate anti clock wide N times. 3) Program to generate sine, full rectifier and square waveforms using DAC. 4) Program to display a given message (max of 8 characters) on seven segment display from left to right and right to left N number of times.	9	
5	**Coprocessor and advance microprocessors:** 8087 architecture, instruction set, programming with 8087, comparison of 8086 with advance microprocessors like 80386,80486, Pentium processors, Core-2 etc.	9	CO3, CO6
	List of experiments 1) Program to perform floating point addition and Subtraction. 2) Program to find square and cube of a floating point number. 3) Program to interface 4*4 HEX keypad. 4) Program to implement elevator.	12	

TEXT BOOKS:

1. The Intel Microprocessor: Architecture, Programming & Interfacing, Barry B. Brey, Pearson Education India, 8 edition, 2014,
2. Microcomputer Systems: 8086/8088 - Family Architecture, Programming and Design, Chen Liu and Glenn A Gibson, Prentice Hall India, 2nd Edition, 2014.

REFERENCE BOOKS:

1. Microprocessors and Interfacing, Douglas V. Hall, , TMH, Revised 2nd Edition, 2006.
2. The Intel Microprocessor Family: Hardware and Software Principles and Applications, James L. Antonakos, Cengage Learning, 2007.
3. Advanced Microprocessors, Daniel Tabak, , TMH, 2nd Edition, 2011

CIE- Continuous Internal Evaluation: Theory (50 Marks)

Bloom's Category	Tests	Assignments	Quizzes
Marks (out of 50)	30	10	10
Remember	10	-	-
Understand	10	5	5
Apply	10	5	5
Analyze	-	-	-
Evaluate	-	-	-
Create	-	-	-

CIE- Continuous Internal Evaluation: Lab (25 Marks)

Bloom's Category	Lab
Marks (out of 25)	
Remember	-
Understand	-
Apply	15
Analyze	10
Evaluate	-
Create	-

SEE- Semester End Examination: Theory (50 Marks)

Bloom's Category	Theory
Marks (out of 50)	
Remember	10
Understand	20
Apply	10
Analyze	10
Evaluate	-
Create	-

SEE- Semester End Examination: Lab (25 Marks)

Bloom's Category Marks (out of 25)	Lab
Remember	-
Understand	05
Apply	15
Analyze	05
Evaluate	-
Create	-

COMPUTER ORGANIZATION

Course Code : 16CSE45 **Credits : 04**
L: P: T: S : 3:0:0:1 **CIE Marks : 50**
Exam Hours : 3 **SEE Marks : 50**

Course Outcomes: At the end of the Course, the Student will be able to:

CO #	COURSE OUTCOME
CO1	Identify the basic functional components of a computer and its interconnection.
CO2	Examine different computer architectures, instruction sets, addressing modes and memory.
CO3	Analyze the cache design parameters and evaluate performance.
CO4	Design and evaluate circuits to perform basic computer arithmetic operations.
CO5	Interpret the working of hardwired and micro-programmed control of the CPU.
CO6	Analyze various ways in which input, output operations are performed.

Mapping of Course Outcomes to Program Outcomes:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	-	-	-	-	-	-	-	-	-	-	-	-
CO2	3	3	-	-	-	-	-	-	-	3	-	-	-	3
CO3	3	3	-	2	-	-	-	-	-	3	-	-	-	3
CO4	3	3	2	2	-	-	-	-	-	-	-	-	2	3
CO5	3	3	2	-	-	-	-	-	-	3	-	-	2	3
CO6	3	3	-	-	-	-	-	-	-	3	-	-	-	3

Course syllabus			
Module No	**Contents of Module**	**Hrs**	**COs**
1	Introduction: Organization and Architecture, Structure and Function of computers, Computer Components, Interconnection structures, Bus interconnection, Performance of computers, Numbers, Arithmetic operations and characters, Memory locations and addresses, CISC and RISC instruction sets, Addressing modes, Stacks, Subroutines	9	CO1, CO2
2	Input/output Organization: Accessing I/O devices, Interrupts, Bus structure, Bus operation, Arbitration	9	CO6
3	Computer Arithmetic: Addition subtraction of signed numbers, Design of fast adders, Multiplication of unsigned and signed numbers, Fast multiplication, Integer Division, Floating point numbers and operations	9	CO4
4	Computer Memory System: Characteristics of Memory System, The Memory hierarchy, Elements of cache design: Cache addresses , Cache size, Mapping function, replacement algorithms, Performance considerations, Semiconductor main memory: Organization, DRAM and SRAM, types of ROM	9	CO2, CO3
5	Basic Processing Unit: Fundamental concepts, Instruction execution, Hardware components, Instruction fetch and execution steps, control signals, hardwired control, CISC style processors	9	CO5

TEXT BOOKS:

1. Computer Organization and Embedded systems, Carl Hamacher, Zvonks Vranesic, Safea Zaky, McGraw Hill, Sixth Edition, 2012.
2. Computer Organization and Architecture, William Stallings, Pearson/PHI, Eighth edition, 2013.

REFERENCE BOOKS:

1. Computer Architecture a quantitative approach, John L. Hennessy and David A. Patterson, Elsevier, Fifth Edition, 2012.
2. Structured Computer Organization, Andrew S. Tanenbaum, PHI/Pearson, Sixth Edition, 2013.
3. Computer Architecture: Fundamentals and principles of Computer Design, Joseph D. Dumas II, BS Publication, 2013.

SELF STUDY

The student shall identify an Emerging topic related to the field of Computer organization and carry out a Self-study on the problem defined. Topic should be socially relevant and research oriented ones. On the completion student will submit a report, which will be evaluated.

CIE- Continuous Internal Evaluation (50 Marks)

Bloom's Category	Tests	Assignments	Quizzes	Self-Study
Marks (out of 50)	30	5	5	10
L1: Remember	20	-	-	-
L2: Understand	10	-	-	-
L3: Apply	-	5	-	5
L4: Analyze	-	-	5	5
L5: Evaluate	-	-	-	-
L6: Create	-	-	-	-

SEE- Semester End Examination (50 Marks)

Bloom's Category	Tests
L1: Remember	20
L2: Understand	10
L3: Apply	10
L4: Analyze	10
L5: Evaluate	-
L6: Create	-

MINI PROJECT

Course Code : 16CSE46 **Credits : 02**
L: P: T: S : 0 : 2 :0 :0 **CIE Marks : 25**
Exam Hours : 3 **SEE Marks : 25**

Course Outcomes: At the end of the Course, the Student will be able to

CO #	COURSE OUTCOME
CO1	Apply the basic object oriented concepts.
CO2	Implement control structures, modularity, I/O and other standard language constructs.
CO3	Implement data abstraction, encapsulation, and inheritance.
CO4	Implement the utilization of abstract interface, polymorphism and standard template library.
CO5	Create a software solution for real time application with proper exception handling.
CO6	Demonstrate their communication skill effectively with technical presentation.

Mapping of Course Outcomes to Program Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	-	-	-	3	-	-	-	3	3	-	3	3	3
CO2	3	3	3	-	3	-	-	-	3	3	-	3	3	-
CO3	3	3	3	-	3	-	-	-	3	3	-	3	3	3
CO4	3	3	3	-	3	-	-	-	3	3	-	3	3	3
CO5	3	3	3	1	3	-	-	2	3	3	-	3	3	-
CO6	3	-	-	-	-	-	-	2	3	3	-	-	-	-

The student shall be capable of identifying a problem related to the field of Computer Science and carry out a mini project on the problem defined. Each student is expected to do the mini project individually. The code developed towards the project will be reviewed by a panel of experts during the course of the semester. Plagiarized projects will automatically get an **"F" GRADE** and the student will be liable for further disciplinary action. At the completion of a project the student will submit a project report, which will be evaluated by duly appointed examiner(s).

CIE- Continuous Internal Evaluation (25 Marks)

Bloom's Category	**Mini Project**
Marks (out of 25)	
Remember	-
Understand	3
Apply	7
Analyze	5
Evaluate	5
Create	5

SEE- Semester End Examination (25 Marks)

Blooms Category	**Tests**
Marks (out of 25)	
Remember	-
Understand	3
Apply	7
Analyze	5
Evaluate	5
Create	5

INTRODUCTION TO ECONOMICS

Course Code: 16HSS421 **Credits: 03**
L:P:T:S : 2:0:0:1 **CIE: 50**
Exam Hour: 03 **SEE: 50**

Course Outcomes: On completion of the course, the student will be able to:

CO1	Gain knowledge about importance of economics in decision making processes in day to day life.
CO2	Analyze business environment at micro and macroeconomic level and its impact on industries in country's economy.
CO3	Acquire knowledge about costing and estimation of projects for profit making.
CO4	Apply principles of budgeting and finance for entrepreneurial success.

Mapping of Course Outcomes to Program Outcomes:

COS	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	1	-	1	-	1	2	2	-	2	2
CO2	2	2	1	-	1	-	1	2	2	-	2	2
CO3	2	2	1	-	1	-	1	2	2	-	2	2
CO4	2	2	1	-	1	-	1	2	2	-	2	2

Module	Contents of Module	Hours	COs
I	**Introduction to Economics**: Role of Engineer as an Economist, Types and problem of economies, Basics of economics (GDP, National income, inflation, business cycle, fiscal and monetary policies, balance of payment).	**4**	**CO1, CO3**
II	**Basic concepts of Microeconomics**: concept of Demand & Elasticity of Demand. Concept of Supply & Elasticity of Supply, Meaning of Production and factors of production, Production Possibility Curve, Law of variable proportions and returns to scale. Relevance of Depreciation towards industry, Depreciation computing methods.	**5**	**CO2, CO3**
III	**Concepts of cost of production**: different types of cost; accounting cost, sunk cost, marginal cost and opportunity cost. Break even analysis, Make or Buy decision. Cost estimation, Elements of cost as Direct Material Costs, Direct Labor Costs, Fixed Over-Heads, Factory cost, Administrative Over-Heads.	**4**	**CO3, CO4**

IV	**Capital budgeting**: Traditional and modern methods, Payback period method, IRR, ARR, NPV, PI. . Interest and Interest factors: Interest rate, Simple interest, Compound interest, Cash - flow diagrams, Personal loans and EMI Payment. Present worth, Future worth.	**4**	**CO1, CO3, CO4**
V	**Book Keeping and Accounts**: Journal, Ledger, Trial balance, asset Types, profit & loss account, balance sheet.	**5**	**CO1, CO2, CO3, CO4**

TEXT BOOKS:

1. Riggs J.L, Engineering Economy, TMH, 2012 edition
2. Jain T.R., Economics for Engineers, VK Publications
3. IM PANDEY,Finacial Management, Vikas Pub. House
4. D N Dwivedi, Mangerial Economics , Vikas Pub. House

REFERENCE BOOKS:

1. Thuesen H.G, Engineering Economy. PHI
2. Prasanna Chandra, Financial Mangement,TMH
3. Singh Seema, Economics for Engineers, IK International
4. Chopra P. N, Principle of Economics, Kalyani Publishers
5. Dewett K K, Modern Economic Theory, S. Chand
6. H. L. Ahuja, Modern Economic Theory, S. Chand
7. Mishra S. K, Modern Micro Economics, Pragathi Publications
8. Gupta Shasi K, Management Accounting, Kalyani Publications

Assessment pattern

CIE –Continuous Internal Evaluation(50 Marks, Theory)

Bloom's category	**Test**	**Assignments**	**SSR**
Marks (out of 50)	**20**	**15**	**15**
Remember	5	-	-
Understand	5	-	-
Apply	5	-	-
Analyze	5	5	5
Evaluate	-	5	5
Create		5	5

SEE –Semester Ending Examination (50 Marks)

Bloom's category	Test
Remember	20
Understand	10
Apply	10
Analyze	10
Evaluate	
Create	

SOFT SKILLS FOR ENGINEERS

Course Code: 16HSS322/16HSS422	**Credits : 3**
L:P:T:S : 2:0:0:1	**CIE Marks : 50**
Exam Hours: 03	**SEE Marks : 50**

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

CO1	Take responsibility for their actions and be accountable to themselves
CO2	Acquire Corporate etiquettes and develop their personality for their professional career
CO3	Understand and learn to manage themselves better and to work with groups
CO4	Set their personal and professional goals by themselves
CO5	Articulate effectively their ideas, thoughts and concepts

Mapping of Course Outcomes to Program Outcomes:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12
CO1	-	-	-	-	-	3	3	3	3	3	-	-
CO2	-	-	-	-	-	3	3	3	3	3	-	-
CO3	-	-	-	-	-	3	3	3	3	3	-	-
CO4	-	-	-	-	-	3	3	3	3	3	-	-
CO5	-	-	-	-	-	3	3	3	3	3	-	-

Module	Contents of the Module	Hours	COs
1.	Taking Ownership, Being Responsible and Accountable for their own actions The meaning of ownership, responsibility and accountability, Practicing these philosophies in	**4**	**CO1**

	everyday life, how do these philosophies build credibility, Developing a 'Credible Character Impression about yourself', Self motivation, Developing healthy Self esteem, Leadership		
2.	**Personality Development and Grooming** Expectations from the industry, building personal presence, corporate grooming, corporate etiquettes, developing personal work code, corporate code of conduct	**10**	**CO2**
3.	**Self Awareness and Self Management** Knowing your own self- understanding personality, perception, values and attitude. Interpersonal skills - Knowing others, working well with others, developing the right attitude for work, being proactive and positive.	**10**	**CO3**
4.	**GOAL Setting** Importance of Goals, Creating SMART goals , Tips for effective execution of goals	**4**	**CO4**
5.	**Articulation and Group Discussion** Ideas generation, expressing thoughts in a logical flow, presenting views in a group	**8**	**CO 5**

Reference Books:

1. The 7 – Habits of Highly Effective People, Stephen R Covey, Neha Publishers.
2. Seven Habits of Highly Effective Teens, Convey Sean, New York, Fireside Publishers, 1998.
3. Emotional Intelligence, Daniel Coleman, Bantam Book, 2006.
4. How to win friends and influence people, Dale Carnegie

Assessment Pattern

Assessment Pattern

CIE- Continuous Internal Evaluation (50 Marks)

Bloom's Category	Tests	Assignments	Quizzes
Marks (out of 50)	30	10	10
Remember	-	-	5
Understand	5	-	5
Apply	10	5	-
Analyze	5	-	-
Evaluate	5	-	-
Create	5	5	-

SEE- Semester End Examination (50 Marks)

Bloom's Category	Tests
Marks (Out of 50)	
Remember	5
Understand	10
Apply	15
Analyze	10
Evaluate	5
Create	5

APPENDIX A

Outcome Based Education

Outcome-based education (OBE) is an educational theory that bases each part of an educational system around goals (outcomes). By the end of the educational experience each student should have achieved the goal. There is no specified style of teaching or assessment in OBE; instead classes, opportunities, and assessments should all help students achieve the specified outcomes.

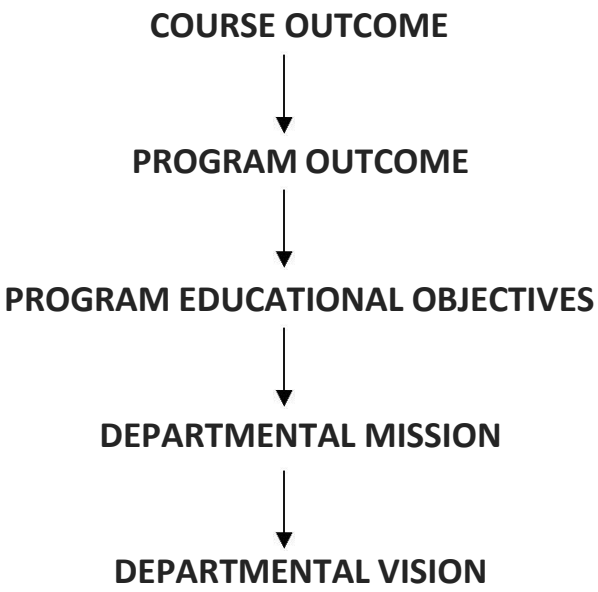
There are three educational Outcomes as defined by the National Board of Accreditation:

Program Educational Objectives: The Educational objectives of an engineering degree program are the statements that describe the expected achievements of graduate in their career and also in particular what the graduates are expected to perform and achieve during the first few years after graduation. [nbaindia.org]

Program Outcomes: What the student would demonstrate upon graduation. Graduate attributes are separately listed in Appendix C

Course Outcome: The specific outcome/s of each course/subject that is a part of the program curriculum. Each subject/course is expected to have a set of Course Outcomes

Mapping of Outcomes



APPENDIX B

The Graduate Attributes of NBA

Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

Problem analysis: Identify, formulate, research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

Conduct investigations of complex problems: The problems that cannot be solved by straightforward application of knowledge, theories and techniques applicable to the engineering discipline that may not have a unique solution. For example, a design problem can be solved in many ways and lead to multiple possible solutions that require consideration of appropriate constraints/requirements not explicitly given in the problem statement (like: cost, power requirement, durability, product life, etc.) which need to be defined (modeled) within appropriate mathematical framework that often require use of modern computational concepts and tools.

Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

Life-long learning: Recognise the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

APPENDIX C

BLOOM'S TAXONOMY

Bloom's taxonomy is a classification system used to define and distinguish different levels of human cognition—i.e., thinking, learning, and understanding. Educators have typically used Bloom's taxonomy to inform or guide the development of assessments (tests and other evaluations of student learning), curriculum (units, lessons, projects, and other learning activities), and instructional methods such as questioning strategies. **[eduglosarry.org]**