Course Curriculum for B. Tech.

in

Computer Science and Engineering

(For students admitted in 2019-20 onwards)

National Institute of Technology
Arunachal Pradesh

P.O. -Yupia, Dist.–Papumpare, Arunachal Pradesh, Pin–791112
Phone No: 0360-2284801/2001582
Fax No: 0360-2284972
www.nitap.ac.in

1.0 Semester wise Credit point distribution

Sl. No.	Year	Credit Point	
		ODD	EVEN
1	First	18.5	20.5
2	Second	23	19
3	Third	21	21
4	Fourth	19	18
Total Credit Point		**77.5**	**82.5**
		160	

1.1 Subject Category wise Credit point Distribution

Course Category	Sem-I	Sem-II	Sem-III	Sem-IV	Sem-V	Sem-VI	Sem-VII	Sem-VIII	Total Credit Point
Core (Basic Science)	10	7	8	--	4	--	--	--	24
Core (Engineering Science)	3	3	--	--	--	--	--	--	9
Core (Professional)	--	2	9	13	13	16	9	--	65
Core (Humanities)	3	3	3	3	--	--	--	--	12
Elective (Professional)	--	--	--	--	--	--	3	--	3
Open Elective	--	--	--	--	--	--	3	--	3
Elective (online course)	--	--	--	--	--	--	--	6	6
Lab (Basic Science)	1	1	--	--	--	--	--	--	0
Lab (Engineering Science)	1.5	2.5	--	--	--	--	--	--	5
Lab (Humanities)	--	--	--	--	--	--	--	--	--
Lab (Professional)	--	2	3	3	3	4	3	--	18
Internship	--	--	--	--	1	1	1	--	3
Academic Project	--	--	--	--	--	--	--	10	10
Audit (NSS/NCC)	0	0	--	--	--	--	--	--	0
Grand Viva	--	--	--	--	--	--	--	2	2
Total Credit Point	18.5	20.5	23	19	21	21	19	18	160

COURSE STRUCTURE

1st Semester						
Sl. No.	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1.	CY-101	Engineering Chemistry	3	0	0	3
2.	CY-102	Engineering Chemistry lab	0	0	2	1
3.	MA-101	Engineering Mathematics I	3	1	0	4
4.	ME-101	Engineering Mechanics	3	0	0	3
5.	ME-102	Workshop Practice I	0	0	3	1.5
6.	MH-101	Communication Skill	0	3	0	3
7.	BIO-101	Bio-Science	3	0	0	3
8.	MH-113	NSS/NCC	0	0	2	0
Total Credits			**12**	**4**	**7**	**18.5**

2nd Semester						
Sl. No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1.	PHY-101	Engineering Physics	3	0	0	3
2.	PHY-102	Engineering Physics Lab	0	0	2	1
3.	MA-102	Engineering Mathematics ll	3	1	0	4
4.	CY-108	Environmental Engineering	3	0	0	3
5.	CSE-112	Introduction to Computer Programming	2	0	0	2
6.	CSE-113	Introduction to Computer Programming Laboratory	0	0	4	2
7.	ME-121	Engineering Drawing	1	0	3	2.5
8.	MH-106	Fundamentals of Economics	3	0	0	3
9.	MH-113	NSS/NCC	0	0	2	0
Total Credits			**15**	**1**	**11**	**20.5**

3rd Semester

Sl. No	Course Code	Course Title	L	T	P	C
1.	EC-203	Digital Logic Design	3	0	0	3
2.	CS-201	Data Structure & Algorithm	3	0	0	3
3.	MA-201	Probability and Statistics	3	1	0	4
4.	MA- 202	Discrete Mathematics	3	1	0	4
5.	MH-201	Introduction to Human values and Ethics	3	0	0	3
6.	CS-202	Object Oriented Programming	3	0	0	3
7.	CS-203	Data Structure & Algorithm Laboratory	0	0	2	1
8.	EC-206	Digital Logic Design Laboratory	0	0	2	1
9.	CS -204	Object Oriented Programming Laboratory	0	0	2	1
Total Credit			**15**	**2**	**6**	**23**

4th Semester

Sl No	Course Code	Course Title	L	T	P	C
1	CS – 206	Computer Organization & Architecture	3	1	0	4
2	CS – 207	Design and Analysis of Algorithm	3	1	0	4
3	CS – 208	Formal Language and Automata Theory	3	0	0	3
4	CS – 209	System Software and administration	2	0	0	2
5	MH – 206	Entrepreneurship	3	0	0	3
6	CS – 210	Computer Organization & Architecture Laboratory	0	0	2	1
7	CS – 211	Design and Analysis of Algorithm Laboratory	0	0	2	1
8	CS – 212	System Software and administration laboratory	0	0	2	1
Total Credits			**14**	**2**	**6**	**19**

5th Semester

Sl. No	Course Code	Course Title	L	T	P	C
1	CS – 301	Operating System	3	1	0	4
2	CS – 302	Compiler Design	3	0	0	3
3	EC – 302	Microprocessor & Interfacing	3	0	0	3
4	MA- 301	Optimization Methods	3	1	0	4
5	CS – 303	Signal and Data Communication	3	0	0	3
6	CS – 304	Operating System Laboratory	0	0	2	1
7	CS – 305	Compiler Design Laboratory	0	0	2	1
8	EC – 306	Microprocessor & Interfacing Laboratory	0	0	2	1
9	CS – 390	**Summer Internship - I**	0	0	0	1
Total Credit			**15**	**3**	**6**	**21**

6th Semester						
Sl. No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	CS – 306	Computer Networking	3	1	0	4
2	CS – 307	Software Engineering	3	0	0	3
3	CS – 308	Computer Graphics and Multimedia	3	0	0	3
4	CS – 309	Database management System	3	0	0	3
5	CS – 310	AI & Machine Learning	3	0	0	3
6	CS – 311	Computer Networking Laboratory	0	0	2	1
7	CS – 312	Database management System Laboratory	0	0	2	1
8	CS – 313	Computer Graphics and Multimedia laboratory	0	0	2	1
9	CS – 314	Machine Learning laboratory	0	0	2	1
10	CS - 391	**Summer Internship –II**	0	0	0	1
Total Credit			**15**	**3**	**6**	**21**

7Th Semester						
Sl. No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	OE-XXX	Open Elective	3	0	0	3
2	CS – 401	Internet and Web Technology	3	0	0	3
3	CS– 402	Cryptography & network Security	3	0	0	3
4	CS– 403X	Elective-I	3	0	0	3
5	CS– 404	Pattern Recognition & Image Processing	3	0	0	3
6	CS – 405	Internet and Web Technology laboratory	0	0	2	1
7	CS – 406	Cryptography & Network Security laboratory	0	0	2	1
8	CS – 407	Pattern Recognition & Image Processing laboratory	0	0	2	1
9	CS– 492	**Summer Internship-III**	0	0	0	1
Total Credit			**15**	**0**	**6**	**19**

8th Semester						
Sl. No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	CS – 408	Elective-II (Swayam/ NPTEL)	3	0	0	3
2	CS– 409	Elective-III (Swayam/ NPTEL)	3	0	0	3
3	CS – 498	Grand Viva	0	0	4	2
4	CS–499	Project Phase-II & Dissertation	0	0	20	10
Total Credits			**9**	**0**	**18**	**18**

List of Subjects for Elective – I

Sl. No	Course Code	Course Title	L	T	P	C
1	CS – 403A	Data Compression	3	0	0	3
2	CS – 403B	Software Project Management	3	0	0	3
3	CS – 403C	Real Time Operating Systems	3	0	0	3
4	CS – 403D	Parallel Algorithms	3	0	0	3
5	CS – 403E	Virtualization and Cloud Computing	3	0	0	3
6	CS – 403F	Soft Computing	3	0	0	3
7	CS – 403G	Natural Language Processing	3	0	0	3
8	CS – 403H	Data Mining & Warehousing	3	0	0	3
9	CS – 403I	Distributed Operating System	3	0	0	3

OPEN ELECTIVE (OFFERED BY OTHER DEPARTMENT)—[OE-XXX]:

- ❖ Students are free to choose any subjects of their interest offered by Electrical Engineering, Electronics and Communication Engineering, Civil Engineering, Mechanical Engineering, Humanities & Management and Basic science during 3rd, 5th and 7th semester.

- ❖ The course has to be of 3 credit course.

ELECTIVE-II and ELECTIVE - III:

- ❖ These two courses will be chosen from NPTEL (https://nptel.ac.in/)/SWAYAM portal (https://swayam.gov.in).

- ❖ Student can enrol in 4th year (7th or 8th semester). However the courses will be credited in 8th semester only.

- ❖ Courses will be of completely student’s choice and should contain at least 20 video lectures including tutorials which will be considered as 3 credit course.

- ❖ The choice of courses should be from outside the core and electives offered by department.

SUMMER INTERNSHIP:

- ❖ At least three internships have to be done in Industry preferably during summer vacation.

Second Semester

2nd Semester						
Sl. No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1.	PHY-101	Engineering Physics	3	0	0	3
2.	PHY-102	Engineering Physics Lab	0	0	2	1
3.	MA-102	Engineering Mathematics ll	3	1	0	4
4.	CY-108	Environmental Engineering	3	0	0	3
5.	CSE-112	Introduction to Computer Programming	2	0	0	2
6.	CSE-113	Introduction to Computer Programming Laboratory	0	0	4	2
7.	ME-121	Engineering Drawing	1	0	3	2.5
8.	MH-106	Fundamentals of Economics	3	0	0	3
9.	MH-113	NSS/NCC	0	0	2	0
Total Credits			**15**	**1**	**11**	**20.5**

Name of the Module: Introduction to Computer Programming

Module Code: CS – 112

Credit Value: 2 {L = 2, T = 0, P =0}

A. Course Objectives:

The course is designed to:

a) Introducing the basic and fundamental components of computers and programming language.
b) Teaching and training of different problems in prior of data structures course.
c) Guiding and training students to write efficient coding,
d) Guiding & training students to fragment problems into different functions or units.

B. Course Content:

Introduction: The von Neumann architecture, machine language, assembly language, high level programming languages, compiler, interpreter, loader, linker, text editors, operating systems, flowchart.
C Fundamentals: Introduction to C, Data types, Constants and variable declaration, Scope, Storage classes, Data input and output functions, Sample programs.
Operators & Expressions: Arithmetic, Relational, Logical, Bitwise operators, Conditional, Assignment, Library functions.

Decision making: Simple if statement, if-else statement, nested if else statement, Switch statement, nested switch, the operator, goto statement.
Decision making & branching: while statement, do-while statement, for statement.

Array: Declaration, Initialization and processing One-dimension array, Two-dimension array and multi dimension array and their operations.

String & pointer: String: Operation on String without using library function and using library function. Pointer: Declaration of pointer variables, accessing the variable by using pointer, pointer increment and decrement operator, pointer and array

Functions: Basic functions, function type, function with no argument & no return value, function with no argument but return value, function with argument & return value, Storage class identifier, Call by reference, Recursive function. Pointer to function.

Structure & Union: Defining a structure, accessing of structure variable, structure and array, array within structure. Nested structure, structure & functions, Pointer & structure, Unions, Enum.

File management system: Advantage of using file, Open, close, read, write in the files, Operation on files.

Dynamic memory Allocation: use of malloc, calloc, realloc, free. Library functions, Implementation of Linked list and their various operations.

The pre-processor: macro statements.

C. Text Books:

1. Kerninghan and Ritchie, The 'C' programming language, 2nd Edition, Pearson, 2008.
2. Yashavant P. Kanetkar, Let Us C, 8th, Infinity Science Press, 2008.
3. Balaguruswamy, Programming In ANSI C, 7thEdition, Tata McGraw-Hill Education, 2017

D. Reference Books:

1. Yashavant P. Kanetkar, ―Let Us C, 16th Edition, BPB Publication, 2017.
2. Zed A. Shaw, Learn C the Hard Way: Pratical Exercises on Computational Subjects You Keep Avoiding (Like C), 2015.
3. Deepali Srivastava and S.K Srivastava, C in Depth, BPB Publication, 2017.
4. Griffiths David and Dawn Griffiths, Head First C, A Brain Friendly Guide, 2012.
5. Grey Perry and Dean Miller, C Programming Absolute Beginner’s Guide, 3rd Edition, 2013.

Name of the Module: Introduction to Computer Programming Laboratory

Module Code: CS - 113

Credit Value: 2 {L = 0, T = 0, P = 4}

A. Course Objectives:

The course is designed to:

b) The student will gain a thorough understanding of the fundamentals of C programming.
c) A student can code, compile and test C programs.
d) Could take Systems programming or Advanced C programming course.
e) Although this course does not deal with object-oriented programming methodology, it will assist the student build the required foundations to undertake a course in OOP..

B. Course Content:

Module l: To write a C program in each case, to find the sum of individual digits of a positive integer, generate the first n terms of the Fibanocci sequence and generate all the prime numbers between 1 and n, where n is a value supplied by the user; to calculate the Sum =1-x 2 /2! +x4 /4!-x 6 /6!+x8 /8!-x 10/10!

Module 2: To write C programs that use both recursive and non-recursive functions, To find the factorial of a given integer and To find the GCD (greatest common divisor) of two given integers; Also, to write a C program, which takes two integer operands and one operator from the user, performs the operation and then prints the result. (Consider the operators +,-,*, /, % and use Switch Statement) and to write a C program that uses functions to perform the Addition of Two Matrices and Multiplication of Two Matrices;

Module 3: To write a C program that uses functions to perform the operations: To insert a sub-string in to a given main string from a given position; To delete n Characters from a given position in a given string; To write a C program to determine if the given string is a palindrome or not; Also to write a C program that displays the position or index in the string S where the string T begins, or – 1 if S doesn’t contain T; To write a C program to count the lines, words and characters in a given text.

Module 4: To write a C program to generate Pascal’s triangle and also to construct a pyramid of numbers; Also to write a C program that uses functions to perform the following operations on singly linked list: Creation, Insertion, Deletion, Traversal;

Module 5: To write C programs that implements stack (its operations) using Arrays, Pointers and that implements Queue (its operations) using Arrays, Pointers;

Module 6: To write a C program that implements the following sorting methods to sort a given list of integers in ascending order using - Bubble sort, Selection sort; Also, to write C programs that use both recursive and non-recursive functions to perform the following searching operations for a Key value in a given list of integers- Linear search, Binary search;

Module 7: To write a C program that implements the following sorting method to sort a given list of integers in ascending order- Quick sort; Also to write a C program that implements the following sorting method to sort a given list of integers in ascending order Merge sort;

C. Text Books:

4. Kerninghan and Ritchie, The 'C' programming language, 2nd Edition, Pearson, 2008.
5. Yashavant P. Kanetkar, Let Us C, 8th, Infinity Science Press, 2008.
6. Balaguruswamy,Programming In ANSI C, 7thEdition, Tata McGraw-Hill Education, 2017

D. Reference Books:

6. Yashavant P. Kanetkar, —Let Us C, 16th Edition, BPB Publication, 2017.
7. Zed A. Shaw, Learn C the Hard Way: Pratical Exercises on Computational Subjects You Keep Avoiding(Like C), 2015.
8. Deepali Srivastava and S.K Srivastava, C in Depth, BPB Publication, 2017.
9. Griffiths David and Dawn Griffiths, Head First C, A Brain Friendly Guide, 2012.
10. Grey Perry and Dean Miller, C Programming Absolute Beginner's Guide, 3rd Edition, 2013.

Third Semester

3rd Semester						
Sl. No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1.	EC-203	Digital Logic Design	3	0	0	3
2.	CS-201	Data Structure & Algorithm	3	0	0	3
3.	MA-201	Probability and Statistics	3	1	0	4
4.	MA- 202	Discrete Mathematics	3	1	0	4
5.	MH-201	Introduction to Human values and Ethics	3	0	0	3
6.	CS-202	Object Oriented Programming	3	0	0	3
7.	CS-203	Data Structure & Algorithm Laboratory	0	0	2	1
8.	EC-206	Digital Logic Design Laboratory	0	0	2	1
9.	CS -204	Object Oriented Programming Laboratory	0	0	2	1
Total Credit			**15**	**2**	**6**	**23**

Name of the Module: Data Structure & Algorithm

Module Code: CS– 201

Credit Value: 3{L = 3, T = 0, P =0}

A. **Course Objectives:**

The course is designed to meet the objectives of:-

a) Designing principles of datastructures and learn the concept of analysis of algorithms.
b) Understand the basic concepts about linear and non-linear data structure and their various operation.
c) Learning essential algorithms for computing.
d) Understanding generic data structures for common engineering problems problem.

B. **Course Content:**

Performance of algorithms: Basic concepts; Mathematical Background; Complexity Analysis; space and time complexity, asymptotics notations, Types of Date Structure.

Linear Data Structures:

Arrays: one dimensional, multi-dimensional, Sparse Matrix, Elementary Operations;
Stacks: Representation, elementary operations and applications such as infix to postfix, postfix evaluation, parenthesis matching;
Queues: Simple queue, circular queue, dequeue, elementary operations and applications;
Linked lists: Linear, circular and doubly linked lists, elementary operations and applications such as polynomial manipulation;

Non-Linear Data Structures:

Trees: Binary tree representation, tree traversal, complete binary tree, heap, binary search tree, height balancedtreeslikeAVLtreeand2-3tree,tries,red-blacktree,Btree,B+tree,m-

waySearchtree,other operations and applications of trees;
Graphs: representation, Adjacencylist, graphtraversal, pathmatrix, connected components, topological sort, Spanning tree, BFS,DFS;

Sorting and Searching:

Sorting: Selection sort, bubble sort, quick sort, merge sort, heap sort, insertion sort, selection sort, radix sort;
Searching: linear and binary search;
Hashing: hash tables, hash functions, open addressing,
File structures: Introduction, data file types, file organization, file access methods.

C. Text Books:

1. S. Sahni, Data Structures, Algorithms, and Applications in C++, Silicon Press, 2/e, 2005.
2. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, Introduction to Algorithms, MIT Press, 3/e, 2009.
3. M. Tenenbaum, Y. Langsam, and M. J. Augenstein, Data Structures Using C and C++, Prentice Hall, 2/e, 1995.
4. Horowitz, Sahni and Anderson-Freed, Fundamentals of Data Structures in C, Silicon Press, 2007.
5. Seymour Lipschutz, Data Structure, The McGrraw Hill, 2013

D. Reference Books:

1. AdamDrozdek,—DataStructuresandAlgorithmsinC++,CengageLearning,2012,4thed..
2. Heileman,—DataStructures,Algorithms&ObjectOrientedProgramming,TataMcgraw -Hill Publishing CompanyLimited.
3. MariappaRadhakrishnan, —Data Structures Using C, BPB Publications,2001.
4. MarkAllenWeiss,—Algorithms,DataStructures,andProblemSolvingwithC++,Addi son- Wesley Publishing Company,1996.
5. HorowitzEllis&SartajSahni,—FundamentalsofDataStructures,GalgotriaPublications
6. AaronM. Tanenbaum,—Data Structures using C, Pearson Education.
7. Ajay Agarwal, —Data structure Through C, Cyber Tech Publications,2005.

Name of the Module: **Object Oriented Programming**

Module Code: CS –202

Credit Value: 3{L = 3, T = 0, P =0}

A. **Course Objectives:**

The course is designed to meet the objectives of:

- **a)** Learning to program in an object-oriented programming language, focusing those who alreadyhave some experience with another programming language, and who now wish to move on to an object-orientedone,
- **b)** Learning object-oriented programming language namely,Java.

B. **Course Content:**

Introduction: Basic features & concepts of Object-Oriented Programming, (OOP), Benefits, Languages and Applications of OOPs.

Java Basics: History of Java, Java buzzwords, datatypes, variables, scope and lifetime of variables, arrays, operators, expressions, control statements, type conversion and costing, simple java program, classes and objects – concepts of classes, objects, constructors, methods, access control, this keyword, garbage collection, overloading methods and constructors, parameter passing, recursion, string handling

Inheritance: Definition, single, multilevel, multiple, hierarchical and hybrid inheritances, virtual base classes, abstract classes

PackagesandInterfaces: Defining,CreatingandAccessingaPackage,Understanding CLASSPATH, importing packages, differences between classes and interfaces, defining an interface, implementing an interface, applying interfaces, variables in an interface and extending interfaces. Exploring packages – Java.io,java.util.

Exception handling and multithreading: Concepts of exception handling, benefits of exception handling, Termination or resumptive models, exception hierarchy, usage of try, catch, throw, throws and finally, built in exceptions, creating own exception subclasses. Differences between multithreading and multitasking, thread life cycle, creating threads, synchronizing threads, daemon threads, threadgroups.

Templates: Class templates, function templates, overloading of function templates, member function templates

Strings: Creating and manipulating string objects, accessing characters in strings.

Applets: Concepts of Applets, differences between applets and applications, the life cycle of an applet, types of applets, creating applets, passing parameters to applets.

Event Handling: Events, Event sources, Event classes, Event Listeners, Delegation event model, handling mouse and keyboard events, Adapter classes, inner classes. The AWT class hierarchy, user interface components- labels, button, canvas, scrollbars, text components, checkbox, checkbox groups, choices, lists panels – scrollpane, dialogues, menubar, graphics, layout manager – layout manager types – border, grid, flow, card and grib bag.

C. Text Books:

1. Java 2: the complete reference, Herbert Schildt, Tata McGraw Hill, 5thEdition
2. Programming with JAVA: a primer, E. Balagurusamy, Tata McGraw Hill, 4th Edition

D. Reference Books:

1. Object-oriented programming through JAVA, V. Vijaya Bhaskar& P. Venkata Subba Reddy, Scitech

Name of the Module: Data Structure & Algorithm Laboratory

Module Code: CS – 203

Credit Value: 1{L = 0, T = 0, P =2}

A. Course Objectives:

1. To demonstrate practical knowledge on Stacks, Queues, Linked lists, Trees Sorting and Hashing Techniques.
2. To analyse suitable data structure to solve real world computing problems.
3. To design solutions for complex computational problems using linear and non-linear data structures.
4. To solve for Complex computational problems by conducting explorative analysis.
5. To use C/C++/Python/Java language for implementing linear and non-linear data structures.
6. To apply contextual knowledge of data structures to design applications for societal requirements.

B. Course Content:

List of Practical:

1. Write the program to understand the numbers of steps for execution of various algorithms.
2. Write the program in Array to demonstrate various operation (eg. Searching: Linear & Binary, Inserting, deleting, two dimensional and multidimensional arrayt, etc).
3. Write the program to implement the various operation in Linked list.
4. Write the program to implement the various operation in Stack and Queue using static and dynamic memory allocation.
5. Write the program to perform the operation(Insert, delete, Traverse, Search. Etc.) in non-linear data structure i.e. Tree.
6. Write the program to perform the various operationsin Graph i.e. BES, DFS, Min. Cost Spanning Tree, All Pairs shortest path etc.
7. Write the program to implement the various sorting techniques (like: Bubble, Quick, Insert, Selection,, etc...)

C. Text Books:

1. S. Sahni, Data Structures, Algorithms, and Applications in C++, Silicon Press, 2/e, 2005.
2. M. Tenenbaum, Y. Langsam, and M. J. Augenstein, Data Structures Using C and C++, Prentice Hall, 2/e, 1995.
3. R. Lafore, Data Structures & Algorithms in Java, 2e, Pearson, 2007
4. GM. T. Goodrich, R. Tamassia, M. H. Goldwasser, Data Structures & Algorithms in Java, 6ed, Wiley, 2014

5. M. T. Goodrich, R. Tamassia, M. H. Goldwasser, Data Structures and Algorithms in Python, Wiley, 2016
6. M. T. Goodrich, R. Tamassia, M. H. Goldwasser, Data Structures and Algorithms in C++, 2ed, Wiley, 2016

D. Reference Books:

1. J Gosling, B Joy, G L Steele and G Bracha, The Java Language Specification, 2/e, Addison-Wesley,2000.
2. B Stroustrup, The C++ Programming Language, 3/e, Addison-Wesley Longman ReadingMA,1997.
3. S B Lippman, C++ Primer, 2/e, Addison-Wesley,1991.
4. T Budd, C++ for Java Programmers, Addison Wesley,1999.
5. M C Daconta, Java for C/C++ programmers, John Wiley & Sons,1996.

Name of the Module: **Object-Oriented Programming Laboratory**

Module Code: CS –204

Credit Value: 1{L = 0, T = 0, P =2}

A. **Course Objectives:**
   a) Identify the importance of object-oriented programming and the variation between structured-oriented programming and object-oriented functions.
   b) To use objects and classes for program creation.
   c) To enhance the capabalities by using numerous object-oriented methods to solve various problems.
   d) Design, develop, test, and debug programs using object-oriented principles in conjuncture with an integrated development environment using Java / C++.

List of practical:

1. Write a program in Java to demonstrate class, constructor, overloading, inheritance, overriding
2. Write a program in Java to demonstrate wrapper class, vectors, arrays
3. Write a program in Java to demonstrate interfaces- multiple inheritances, extending interfaces
4. Write a program in Java to demonstrate packages
5. Write a program in Java to demonstrate multithreaded programming, handling errors and exceptions, applet programming and graphics programming
6. Write a program in Java to demonstrate Java SWING application.
7. Write a program in Java to demonstrate Client-Server Programming.

Reference Books:

(i) Java 2: the complete reference, Herbert Schildt, Tata McGraw Hill, 5th Edition
(ii) Programming with JAVA : a primer, E. Balagurusamy, Tata McGraw Hill4th Edition
(iii) Object oriented programming through JAVA, V. VijayaBhaskar& P. VenkataSubba Reddy, Scitech
(iv) Teach yourself C++ , Herbert Schildt, Tata McGraw Hill
(v) Object Oriented Programming with C++, E. Balagurusamy, Tata McGraw Hill, 6th Edition

Forth Semester

4th Semester						
Sl No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	CS – 206	Computer Organization & Architecture	3	1	0	4
2	CS – 207	Design and Analysis of Algorithm	3	1	0	4
3	CS – 208	Formal Language and Automata Theory	3	0	0	3
4	CS – 209	System Software and administration	2	0	0	2
5	MH – 206	Entrepreneurship	3	0	0	3
6	CS – 210	Computer Organization & Architecture Laboratory	0	0	2	1
7	CS – 211	Design and Analysis of Algorithm Laboratory	0	0	2	1
8	CS – 212	System Software and administration laboratory	0	0	2	1
		Total Credits	**14**	**2**	**6**	**19**

Name of the Module: Computer Organization & Architecture

Module Code: CS – 206

Credit Value: 4{L = 3, T = 1, P =0}

A. Course Objectives:

The course is designed to meet the objectives of

1. Helping the students to develop an understand the nature and characteristics of the organisation and design of the modern computer systems,
2. Focusing on the organisation & operation of the CPU

B. Course Content:

Fundamentals of Computers: Digital computers, layers in computer system, types of computers, history of computers

Data representation and computer arithmetic: Data types, complement, fixed point representation, floating point representation, multiplication and division of sign and unsigned integers.

Micro operation and design of arithmetic logic unit: Register transfer micro operation, bus transfer, memory transfer, arithmetic micro operation, logic micro operation, logic unit, shift unit, design of arithmetic and logic unit.

Instruction set: Instruction code, register, computer instruction, timing and control, instruction cycle, instruction formats, CPU organization, instruction length, addressing standard, addressing mode, instruction set, RISC, CISC.

Design of control unit: hardware control design, micro programmed control.

Memory organization: memory hierarchy, main memory, cache memory, virtual memory.

Input-output organization: peripheral device, I/O interface and I/O driver, synchronous and asynchronous data transfer, modes of data transfer, priority interrupt, DMA, input- output processor.

Parallel processing: performance measurement of computer, parallel computer structure, general classification of computer architecture, pipelining, vector processing, multiprocessor system, flow computers.

C. Text Books:

1. W. Stallings, Computer Organization and Architecture: Designing for Performance, 8th Ed., Pearson Education India.2010.
2. D. A. Patterson and J. L. Hennessy, Computer Organization and Design, 4th Ed., Morgan Kaufmann, 2008.
3. A. S. Tanenbaum, Structured Computer Organization, 5th Ed., Prentice Hall of India,2009.

D. Reference Books:

1. M.M.Mano, ―ComputerSystemArchitecture|, Pearson Education.
2. C. Hamacher, Z. Vranesic, ―Computer Organisation‖, Tata McgrawHill, 2011.
3. M. Jain, S.Jain, V. Pillai, ―ComputerOrganizationandSystemSoftware|, BPBPublications, 2003.
4. P.PalChaudhuri,―ComputerOrganisation&Design‖,PHILearningPrivateLtd.,2009.
5. J. P.Hayes, ―Computer Architecture & Organisation‖, McGraw Hill,1998
6. T.K.Ghosh&A.J.Pal,―ComputerOrganization&Architecture‖,TataMcGraw-Hill,
7. M. Rafiquzzaman,‖Computer Architecture|, Prentice HallofIndia.

Name of the Module: Design & Analysis of Algorithm

Module Code: CS – 207

Credit Value: 4{L = 3, T = 1, P =0}

A. Course Objectives:

The course is designed to meet the objectives of:

1. Ability to analyze the performance of algorithms.
2. Ability to choose appropriate algorithm design techniques for solving problems.
3. Ability to understand how the choice of data structures and the algorithm design methods impact the performance of programs.
4. To clear up troubles the usage of set of rules design methods including the grasping approach, divide and overcome, dynamic programming, backtracking and department and certain.
5. To understand the variations among tractable and intractable problems.
6. To introduce p and np classes.

B. Course Content:

Models of computation: RAM, TM etc. time and space complexity

Asymptotic Notation: Big-O, omega, theta etc.; finding time complexity ofwell known algorithms like- heap sort, search algorithm etc.
AlgorithmDesigntechniques: Recursion-Definition,Use,Limitations,andExamples:Hanoiproblem. Tail Recursion, etc.

Divide and Conquer: Basic method, use, Examples: Merge sort, Quick Sort, Binary Search, etc.

Dynamic Programming: Basic method, use, Examples: matrix-chain multiplication, all pair shortest paths, single-source shortest path, travelling Salesman problem, etc.

Branch and Bound: Basic method, use, Examples: The 15-puzzle problem, etc.

Backtracking: Basic method, use, Examples: Eight queens problem, Graph coloring problem, and Hamiltonian problem, etc.

Greedy Method: Basic method, use, Examples: Knapsack problem, Job sequencing with deadlines, minimum spanning tree (Prim's and Kruskal's algorithms), etc.
Lower Bound Theory: Bounds on sorting and sorting techniques using partial and total orders.
Disjoint Set Manipulation: Set manipulation algorithm like UNION-FIND, union by rank, Path compression.
Matrix manipulation algorithms: Different types of algorithms and solution of simultaneous equations, DFT & FFT algorithm; integer multiplication schemes

Notion of NP-completeness: Non deterministic algorithm, COOK‘s theorem, P class, NP-hard class, NP-complete class, CNF Satisfiability problem, proof a problem to be NP hard, Clique Decision Problem.
Approximation algorithms: Necessity of approximation scheme, performance guarantee, Polynomial time approximation schemes: 0/1 knapsack problem
String Matching Algorithm

C. Text Books:

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, Introduction to Algorithms, MIT Press, 3/e, 2009.
2. M. T. Goodrich and R.Tamassia, Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition, Wiley, 2006.
3. E. Horowitz, S. Sahni, Fundamentals of Computer Algorithms, 2e, Universities Press, 2008.

D. Reference Books:

1. K. Mehlhorn , ―Data Structures and Algorithms 2: Graph Algorithms and NP-Completeness, Springer Science & Business Media, 2012.
2. S.Baase ―Computer algorithms : introduction to design and analysis‖, Pearson Education India, 2009.
3. E.Horowitz and S. Shani ―Fundamentals of Computer algorithms‖, Galgotia Publications, 1984.
4. Edward Martin Reingold, J urgNievergelt, N. Deo,―Combinational algorithms: Theory and Practice, Pearson Education Canada, 1977.
5. Allan Borodin, Ian Munro, ―The computational complexity of Algebraic and Numeric problems, American Elsevier Pub.Co., 1975.
6. Steven S Skiena,―The Algorithm Design Manual‖, Springer Science & Business Media, 2009, 2, illustrated, reprint.
7. Alfred V Aho, John E Hopcroft,―Design and Analysis of Computer Algorithms‖, Pearson Education India, 1974, ISBN 8131702057, 9788131702055

Name of the Module: Formal Language & Automata Theory

Module Code: CS – 208

Credit Value: 3{L = 3, T = 0, P =0}

A. Course Objectives:

The course is designed to meet with the objectives of:

1. providing a deeper understanding of programming languages design motivations and semantics, facilitating students to select and use the most appropriate language for a given task and write correct programs,
2. Illustrating language processing techniques: compilation and interpretation.

B. Course Content:

Instruction to the theory of formal languages, Chomsky Hierarchy of languages, definition, recognition of a language by an automata, grammar, DFA, NFA, equivalence of DFA and NFA, regular sets & regular expressions, equivalence of Moore & Mealy machines, applications of finite automata.

Closure Properties of Regular Sets : Pumping lemma & its application, closure properties minimization of finite automata: minimisation by distinguishable pair, Myhill-Nerode theorem.

Context Free Grammars: Introduction, definition, derivation trees, simplification, CNF & GNF.

Pushdown Automata: Definition, moves, Instantaneous Descriptions, language recognized by PDA, acceptance by final state & empty stack, the equivalence of PDA and CFG, deterministic PDA.

ClosurePropertiesofCFLs: Pumpinglemma&itsapplications,closureproperties,decision algorithms.

Turing machine: Informal proofs that some computational problems cannot be solved, Turing machines (TMs), their instantaneous descriptions. Language acceptance by TMs. Hennie convention for TM transition diagrams, halting problem of TM, Recursively enumerable (r.e.) and recursive languages, notion of undecidable problems. Universal language and universal, some undecidable problems of TMs. Rice's theorem.

C. Text Books:

1. ShyamalenduKandar, Introduction to Automata Theory, Formal Languages and Computation, Pearson Education India, 2013
2. J. E. Hopcroft and J. D. Ullman, Introduction to Automata Theory, Languages & Computation, 2nd Edition, Pearson education.
3. K.L. P Mishra & N. Chandrasekharan, Theory of Computer Science: Automata, Languages and Computation,3rd Edition, Prentice Hall India.
4. DasaradhRamaiah K., Introduction to Automata and Compiler Design, Prentice Hall India Learning Private Limited

D. Reference Books:

1. Richard Y. Kain, "Theory ofAutomata& Formal Language", McGrawHill.
2. ZviKohavi, "Switching and Finite Automata Theory", 2nd edition, TataMcGraw-Hill.
3. Peter Linz, "An Introduction to Formal Languages and Automata", Jones & Bartlett Publishers.
4. Howard Straubing, "Finite Automata, Formal Logic, and Circuit Complexity", Springer.
5. John Carroll and Darrell Long, "Theory of Finite Automata:With an Introduction to Formal Languages", Prentice Hall,1989.

Name of the Module: System Software & Administration

Module Code: CS – 209

Credit Value: 2{L = 2, T = 0, P =0}

A. **Course Objectives:**
B. Students completing this course will be able to
    1. Know/understand the working principle of compilers, interpreters, linkers, loaders and hence, be able to create new ones.
    2. Understand and create system softwares like firmware, bootloader, kernel, drivers, kernel modules, filesystems and system libraries.
    3. Know the instruction set of 8085/8086/80386 microprocessor and be able to make programs for them with assembly language programming.
    4. Operate on the command-line interface with shell scripting.
    5. Administrate a Unix/Linux-based system.
    6. Administrate a Windows/NT system
    7. Develop software in a collaborative manner and manage it with distributed CVS.
    8. Write documentation for developed software.

C. **Course Content**

Basic overview of a computer system: firmware, bootloader, kernel, userspace, Functions of firmware. Types of firmware. Difference between kernel space and user space. Difference between monolithic kernel and microkernel. Functions of kernel modules. Functions of a driver.

Assemblers: General design procedures, instructions and data representations; Design of two pass assemblers; Linux assembly language, assembly language programming and simulation ofx86 architecture; cross Assemblers

Macro Processors: Features of a macro facility, macro instruction arguments, conditional macro expansion, macro calls within macros, Macro Assemblers.

Loader schemes: Compile and go loaders, absolute loaders, relocating loader, linking, Reallocation: static & dynamic linking. Direct linking loaders, Binders, overlays, dynamic binders; working principle of debuggers.

Overview of Unix system, commands and utilities;
1. Linux installation from scratch.
2. Study of file systems and directory layout.
3. Using package managers like dnf/rpm, apt/dpkg, pacman, etc.
4. Basic Linux utilities, logging, backup, authentication.
5. Text editors: vim, emacs, nano, etc.
6. Software development: make, sccs, git, gdb.
7. Archiving: tar, zip, 7zip, gzip, bzip2, xz, etc. Shell: bash, tcsh, zsh, etc.
8. Shell environment and configuration. Regex: grep., egrep, sed, awk, etc.

9. Scripting languages: Python, JavaScript, PHP, Perl.
10. Internet mail system: sendmail, dovecot, postfix, etc.
11. Web app development: PHP, MariaDB.
12. Daemons and services: startup targets, runlevels, service configuration, service management with init system. Case study of services like DNS, HTTP, IMAP/SMTP, DNS.

Windows Administration: Managing the server operating system, file and directory services, software distribution and updates, profiling and monitoring assigned servers, security and troubleshooting;

Documentation and Presentation: LaTeX, Markdown, etc.

Software development and collaboration: Usage of distributed CVS hosting: GitHub, GitLab, Bitbucket, etc. Issue trackers: Jira, Trac, etc. CI/CD: Jenkins, Travis, etc.

D. Text Books:

1. Andrew S. Tanenbaum, Todd Austin, ―Structured Computer Organization, Pearson, 2013, 6th Edition, ISBN: 0132916525,9780132916523
2. Robert L. Britton, ―MIPS Assembly Language Programming, Pearson/Prentice Hall, 2004, ISBN: 0131420445,9780131420441
3. L.L. Beck, ―System Software, (3rd Ed.), Pearson Education India, 1997, ISBN: 817758555X,9788177585551
4. L. Lamport, ―LaTeX: A Document Preparation System, 2nd Ed., Addison-Wesley Series, 1994.

E. Reference Books:

1. Donovan, ―Systems Programming, Tata McGraw-Hill Education, 2001, ISBN: 0074604821.
2. Dandamudi, Sivarama P., ―Guide to Assembly Language Programming in Linux, Springer US,2005, eBook ISBN:978-0-387-26171-3
3. The Web Technologies Series, Cengage Learning, 2010, ISBN:0538745843, 9780538745840
4. B. Kauler, ―Windows assembly language & Systems Programming: 16- and 32-Bit Low- Level Programming for the PC and Windows, 2nd Ed., CMP Books; August1997
5. S. Kochan and P. Wood, ―Unix Shell programming, 3rd Ed., SAMS, 2003.
6. S. Das, ―Unix System V.4 Concepts and Applications, 3rd Ed., Tata Mcgraw-Hill,2003.
7. Nithyashri, ―System Software, Tata McGraw-Hill Education, 2010, ISBN: 0070671923, 9780070671928.
8. A.A.PuntambekarI.A.Dhotre, ―System Software, Technical Publications, 2007,ISBN: 8184310307, 9788184310306
9. ShantanuChattopadhyay, ―System Software, PHI Learning Pvt. Ltd., 2007, ISBN: 812033051X,9788120330511.
10. Kirch, ―Linux network Administrator‗s guide (2nd Ed.),O‗Rielly
11. Steve Maxwell, ―UNIX System Administration: A Beginner's Guide, McGraw

Hill Professional, 2002.

12. Limoncelli, ―The Practice of System & Network Administration,Pearson

Name of the Module: Computer Organization & Architecture Laboratory
Module Code: CSE - 210
Credit Value: 1{L=0, T=0, P=2}

List of practical:

1 Write a verliog code to implement S-R flipflop and also implement the testbench to verify the outputs.

2 Write a verliog code to implement D- latch and also implement the testbench to verify the outputs.

3 Write a verliog code to implement D- latch with reset asynchronous and also implement the testbench to verify the outputs.

4 Write a verliog code to implement D- latch with reset synchronous and also implement the testbench to verify the outputs.

5 Write a verliog code to T- flipflop with asynchronous reset and also implement the testbench to verify the outputs.

6 Write a verliog code to T- flipflop with synchronous reset and also implement the testbench to verify the outputs.

7 Write a verliog code to implement JK flipflop and also implement the testbench to verify the outputs.

8 Write a verliog code to implement Counter also implement the testbench to verify the outputs.

9 Write a verliog code to implement 8 bit Ripple carry adder and also implement the testbench to verify the outputs.

10 Write a verliog code to design an ALU and also implement the testbench to verify the outputs.

Books:

1. Verilog HDL: A Guide to Digital Design and Synthesis, Samir Palnitkar, 2nd edition, Pearson.
2. A Verilog HDL Primer, Jayaram Bhaskar, 3rd edition, Star Galaxy Publishing.
3. The Verilog Hardware Description Language, Donald E. Thomas, Philip R. Moorby, 5th edition, Springer.

Name of the Module: Design & Analysis of Algorithm Laboratory
Module Code: CSE - 211
Credit Value: 1 {L=0, T=0, P=2}

A. Course Objectives:

1. Ability to write programs in C/C++/Python/Java to solve the various realistic and engineering problems using Divide and Conquer paradigm.
2. Ability to write programs in C/C++/Python/Java to solve the various realistic and engineering problems using Greedy paradigm.
3. Ability to write programs in C/C++/Python/Java to solve the variousrealistic and engineering problems using Dynamic programming.
4. Ability to write programs in C/C++/Python/Java to solve the variousrealistic and engineering problems using backtracking Strategies.
5. Able to Write a Dynamic Programming algorithm for the 0/1 Knapsack problem and also Dynamic Programming algorithm for the Optimal Binary Search Tree Problem.

B. Course Content:

List of practical: (Minimum five experiments should be conducted by students)

1. Write the programs for solving Binary Search; Merge Sort; Quick Sort; Find Maximum and Minimum element from an array of integer, etc…in Divide and Conquer Strategy.

2. Write the programs for solving the minimum number of scalar multiplication needed for chain of matrix; all pair of Shortest path for a graph (Floyed Warshall Algorithm);Single Source shortest Path for a graph (Dijkstra, Bellman Ford); Traveling Salesman Problem, etc… in Dynamic Programming.

3. Write the programs for solving the 15 Puzzle Problems, etc… in branch and bound.

4. Write the programs forsolving8 Queen Problem; Graph Coloring Problem; Hamiltonian Problem in Backtracking.

5. Write the programs for solving Knapsack Problem; Job sequencing with deadlines; Minimum Cost Spanning Tree by Prim's Algorithm; Minimum Cost Spanning Tree by Kruskal's Algorithm; etc…in Greedy Paradigm.

6. Write the programs for solving Breadth First Search (BFS); Depth First Search (DFS); etc. in Graph Traversal Algorithm.

C. Text Books:

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, Introduction to Algorithms, MIT Press, 3/e, 2009. Wiley, 2006.
2. E. Horowitz, S. Sahni, Fundamentals of Computer Algorithms, 2e, Universities Press, 2008.

D. Reference Books:

1. J Gosling, B Joy, G L Steele and G Bracha, The Java Language Specification, 2/e, Addison-Wesley,2000.
2. B Stroustrup, The C++ Programming Language, 3/e, Addison-Wesley Longman ReadingMA,1997.
3. S B Lippman, C++ Primer, 2/e, Addison-Wesley,1991.
4. T Budd, C++ for Java Programmers, Addison Wesley,1999.
5. M C Daconta, Java for C/C++ programmers, John Wiley & Sons,1996.

Name of the Module: System Software & Administration Laboratory
Module Code: CS - 212
Credit Value: 2 {L=0, T=0, P=2}

List of practical:

1. Study and installation of Unix based operating system.
2. Study and practices of basic Unix commands, shortcuts and pipelining, managing accounts, privileges given to different users, create groups, changes passwords, etc.
3. Introduction of Shell Programming withexamples.
4. Study and practice of Linux assembly language programming using x86 assembly language programming.
5. Study and practice of NASM assembly languageprogramming.
6. Implementation of Pass 1 and Pass 2 of aassembler.
7. Implementation of Macroprocessor.
8. Implementation a Symbol Table With differentfunctions.
9. Implementation of a Single Pass and two pass MacroProcessor.
10. Study and configuration of NFS/HTTP/mail/DNS.
11. Study and practice of Process Identification. Creation and Kill using system call fork() &exec() functions and Zombieprocesses.
12. Software development with distributed CVS: make, git, GitHub, etc.
13. Writing documentation with LaTeX.
14. Study of a CI/CD pipeline on Jenkins.

Proposed Courses for Fifth Semester

5th Semester						
Sl. No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	CS – 301	Operating System	3	1	0	4
2	CS – 302	Compiler Design	3	0	0	3
3	EC – 302	Microprocessor & Interfacing	3	0	0	3
4	MA- 301	Optimization Methods	3	1	0	4
5	CS – 303	Signal and Data Communication	3	0	0	3
6	CS – 304	Operating System Laboratory	0	0	2	1
7	CS – 305	Compiler Design Laboratory	0	0	2	1
8	EC – 306	Microprocessor & Interfacing Laboratory	0	0	2	1
9	CS – 390	**Summer Internship - I**	0	0	0	1
		Total Credit	**15**	**3**	**6**	**21**

Name of the Module: Operating System
Module Code: CS - 301
Credit Value: 4 {L=3, T=1, P=0}

A. Course Objectives:

The course is designed to meet the objectives of:

1. Appreciating the role of an operating system,
2. Making aware of the issues in management of resources like processor, memory and input- output.
3. Selecting appropriate productivity enhancing tools or utilities for specific needs like filters or version control.
4. Obtaining some insight into the design of an operating system.

B. Course Content:

Introduction: Introduction to Operating System: Operating system functions, evaluation of Operating System, Different types of Operating System: batch, multi-programmed, time-sharing, real-time, distributed, parallel.

System Structure: Computer system operation, I/O structure, storage structure, storage hierarchy, different types of protections, operating system structure (simple, layered, virtual machine), Operating System services, system calls.

Process Management: Processes: Concept of processes, process scheduling, operations on processes, co-operating processes, inter- process communication. Threads: overview, benefits of threads, user and kernel threads.

CPU Scheduling: Scheduling criteria: pre-emptive & non-pre-emptive scheduling, scheduling algorithms (FCFS, SJF, RR, priority), algorithm evaluation, multi-processor scheduling.

Process Synchronization: background, critical section problem, critical region, synchronization hardware, classical problems of synchronization, semaphores.

Deadlocks: system model, deadlock characterization, methods for handling deadlocks, deadlock prevention, deadlock avoidance, deadlock detection, recovery from deadlock.

Storage Management: Memory Management: background, logical vs. physical addresses space, TLB, swapping, contiguous memory allocation, paging, segmentation, segmentation with paging. Virtual Memory: background, demand paging, performance, page replacement, page replacement algorithms (FCFS, LRU), allocation of frames, thrashing.

File Systems: file concept, access methods, directory structure, file system structure, allocation methods (contiguous, linked, indexed), free-space management (bit vector, linked list, grouping), directory implementation (linear list, hash table), efficiency & performance.

I/O Management: I/O hardware, polling, interrupts, DMA, application I/O interface (block and character devices, network devices, clocks and timers, blocking and non blocking I/O), kernel I/O subsystem (scheduling, buffering, caching, spooling and device reservation, error handling), performance.

Disk Management: diskstructure, disk scheduling (FCFS, SSTF, SCAN, C-SCAN), disk reliability, disk formatting, boot block, bad blocks.

Protection & Security: Goals of protection, domain of protection, security problem, authentication, one time password, program threats, system threats, threat monitoring, encryption.

Case Studies; Dos & Unix.

C. Text Books:

1. Abraham Silberschatz, Peter B.Galvin, Greg Gagne, Operating System Concepts, Wiley, 2014, 9th ed., illustrated, revised.
2. Andrew S Tanenbaum, Albert S Woodhull, Operating System Design & Implementation, Pearson Education, 2011, 3rd ed.
3. M. Dhamdhere, Operating Systems: A Concept-based Approach, Tata McGraw-Hill Education, 2006, 2nded.

D. Reference Books:

1. William Stallings, Operating Systems, Pearson Education India, 2006, 5thed..
2. Harvey M. Deitel, An Introduction to Operating Systems, Addison-Wesley, 1990, 2, illustrateded.
3. Maurice J. Bach,Design of the Unix Operating System, Prentice-Hall, 1986 ,15, illustrated.
4. Milenkovie M., Operating System: Concept & Design, Tata McGraw-Hill Education, 2001, 2nded.

Name of the Module: Compiler Design
Module Code: CS - 302
Credit Value: 3 {L=3, T=0, P=0}

A. Course Objectives:

The course is designed to meet with the objectives of

a. Providing a thorough introduction to the theory and practice of programming language translation.
b. Introducing to the design and implementation of programming language translators.

B. Course Content:

Compilers, Analysis of the source program: The phases of the compiler.

The role of the lexical analyzer: Tokens, Patterns, Lexemes, Input buffering, Specifications of token, Recognition of a tokens, Finite automata, Regular expression to Finite Automata, Design of a lexical analyzer generator (LEX).

The role of a parser: Context free grammars, Writing a grammar, Top down Parsing, Non- recursive Predictive parsing (LL), Bottom up parsing, Handles, Viable prefixes, Operator precedence parsing, LR parsers (SLR, LALR), Parser generators (YACC). Error recovery strategies for different parsing techniques.

Syntax director definitions: Construction of syntax trees, Bottom-up evaluation of S attributed definitions, L attributed definitions, Bottom-up evaluation of inherited attributes. Type systems, Specification of a simple type checker, Equivalence of type expressions, Type conversions.

Sourcelanguageissues(Activationtrees,Controlstack,scopeofdeclaration,Bindingofnames), Storage organization (Subdivision of run-time memory, Activation records), Storage allocation strategies,Parameterpassing(callbyvalue,callbyreference,copyrestore,callbyname),Symbol tables, dynamic storage allocationtechniques.

Intermediate languages: Graphical representation, Three-address code, Implementation of three address statements (Quadruples, Triples, Indirect triples).

Code optimization: Introduction, Basic blocks & flow graphs, Transformation of basic blocks, DAG representation of basic blocks, The principle sources of optimization, Loops in flow graph, Peephole optimization.

Issues in the design of code generator:a simple code generator, Register allocation & assignment.

C. Text Books:

1. Aho,Sethi,Ullman-—CompilerPrinciples,TechniquesandTools-PearsonEducation,ISBN 8131759024,9788131759028.
2. Holub-—CompilerDesigninC,Prentice-HallofIndiaPvt.Limited,2006,ISBN812030778X, 9788120307780.

D. Reference Books:

3. Keith Cooper, Linda Torczon, Cooper and Torczon. Engineering a compiler,Elsevier, 2011, ISBN 0080916619,9780080916613.
4. V. Raghavan, Principles of Compiler Design, McGrawHill,2010.

Name of the Module: Signal and Data Communication
Module Code: CS - 303
Credit Value: 3 [L=3, T=0, P=0]

A. Course Objectives:

1. To have a detailed study of various analog and digital modulation and demodulation techniques.
2. To have a thorough knowledge of various multiplexing schemes and Data communication protocols.
3. To know about the standards of the communication.

B. Course Content:

Introduction, Continuous-time and discrete-time signals, linear system, system properties,
Input-output modeling using linear differential equations, Time –varying systems,
Convolution representation of LTI and discrete and continuous signals, linear time-varying systems
Signals in terms of frequency components, fourier transform as limiting form of fourier series, properties, response to sinusoidal, periodic and aperiodic inputs, sampling analog modulation and demodulation of signals, simultaneous transmission of signals, digital modulation
Analog and digital data transmission, transmission impairments, channel capacity
Wire and wireless transmission, signal encoding techniques, FDM, synchronous and sat TDM, ADSL, XDSL
Spread spectrum analysis (FHSS, DSSS, OFDM), error handling, introduction X.25, frame Relay

C. Text Books:

1. *V. Oppenheim, A. S. Willsky and A. H. Nawab: Signals and Systems;PHI*
2. *W. Stallings: Data and Computer Communications; PearsonEd*
3. *S. Haykin and B. V. Veen: Signals and Systems;Willey.*

D. Reference Books:

1. *Das: Digital Communication: Principles and System Modelling;Springer.*
2. *Tarun Kumar Rawat: Signals and system, Oxford*
3. *R. L. Freeman: Practical Data Communications; Willey.*
4. *M. S. Roden: Digital and Data Communication Systems; PrenticeHall.*
5. *Ahmad: Data Communication Principles: For Fixed and Wireless Networks;Kluwer.*

Name of the Module: Operating System Laboratory
Module Code: CS - 304
Credit Value: 1 {L=0, T=0, P=2}

List of practical:

1. Basics of UNIX commands.
2. Shell programming
3. Implementation of CPU scheduling. a) Round Robin b) SJF c) FCFS d)Priority
4. Implement all file allocationstrategies
5. Implement Semaphores
6. Implement ll File OrganizationTechniques
7. Implement Bankers algorithm for Dead LockAvoidance
8. Implement an Algorithm for Dead LockDetection
9. Implement the all page replacement algorithms a) FIFO b) LRU c)LFU
10. Implement Shared memory andIPC
11. Implement Paging Technique f memorymanagement.
12. Implement Threading &SynchronizationApplications

Name of the Module: Compiler Design Laboratory
Module Code: CS - 305
Credit Value: 1 {L=0, T=0, P=2}

Programming assignments related to

1. Divide the given input program into lexemes.
2. Compute FIRST, FOLLOW function.
3. Implement operator precedence parsing.
4. Implement recursive descent parsing.
5. Implement lexical analyzer using LEX tool.
6. Use of LEX program to identify a simple and a compound statement
7. Count the number of keywords and identifiers in a sentence
8. YACC program to check the validity of an arithmetic expression.
9. An assignments to design a small compiler.

Proposed Courses for Sixth Semester

6th Semester						
Sl. No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	CS – 306	Computer Networking	3	1	0	4
2	CS – 307	Software Engineering	3	0	0	3
3	CS – 308	Computer Graphics and Multimedia	3	0	0	3
4	CS – 309	Database management System	3	0	0	3
5	CS – 310	AI & Machine Learning	3	0	0	3
6	CS – 311	Computer Networking Laboratory	0	0	2	1
7	CS – 312	Database management System Laboratory	0	0	2	1
8	CS – 313	Computer Graphics and Multimedia Laboratory	0	0	2	1
9	CS – 314	Machine Learning laboratory	0	0	2	1
10	CS - 391	**Summer Internship –II**	0	0	0	1
Total Credit			**15**	**3**	**6**	**21**

Module: Computer Networks

Code: CS - 306

Credit Value: 4 {L=3, T=1, P=0}

A. Course Objectives:

1. The course is designed to meet with the objectives of:
    1. Understanding the state-of-the-art in network protocols, architectures, and applications.
    2. Examining and studying of different protocols in OSI and TCP/IP.
    3. Understanding of network addressing, mapping etc.
    4. Understanding error control flows control packet recovery etc.
    5. Understanding the structure of LAN, WAN and MAN.
    6. Understanding internetworking of devices.

B. Course Content:

Basic: Introduction to Networking and its origin, layered task, Protocol stack, OSI model, TCP/IP model and brief functionality.

Physical layer and media: Data, Signals, Transmission, Digital transmission- digital to digital conversion, Analog to digital conversion, bandwidth utilization and spread spectrum.

Circuit and Packet Switching:- Switched Networks, Circuit-Switching Networks, Switching Concepts, Routing in Circuit-Switched Networks, Control Signalling, Packet-Switching Principles, Routing, Congestion Control, X.25 282. , structure of a switch.

Data link layer: Error correction and Detection, Data link control- framing, flow and error control, Noise less channels- Simple Protocols, Stop and wait protocol, Noisy channel protocol- Stop and Wait ARQ, Go and Back N ARQ, Selective Repeat AutomaticRepeatRequest,HDLC-ConfigurationandTransfermode,MultipleAccess-Random Access, Control access, Channelization, Wired Network (IEEE 802.3), Wireless

Network (IEEE 802.11), Virtual LAN, Virtual Circuit Networks-Frame relay and ATM LANetc.

Network Layer: Logical Addressing, Internet Protocol (IP), Address mapping, Error reporting, and multicasting- ARP, RARP, BOOTP, DHCP, ICMP,IGMP , Network Address Translators (NAT) Network Delivery-Delivery, Forwarding and Routing, Unicast routing protocol- Intra & inter domain routing, RIP, OSPF, BGP

Transportlayer: ProcesstoProcessdelivery-Connectionorientedandconnectionless service, UDP, TCP, SCTP, error and flow controls, Congestion control and Quality of service-Open loop congestion control, Closed loop congestion control, Congestion control in TCP and in frame relay Quality of service-flow characteristics, flow cases, different techniques to improve QoS, RSVP.

Application layer: Name Space, Domain in Namespace, Distribution of name space, DNS- generic, country and inverse domain, Resolution: Resolver, Mapping name to Address, Mapping address to names, recursive resolution. Remote logging- telnet, Electronic mail-SMTP, POP, IMAP and file transfer- FTP architecture, commands of FTP.WWWandHTML-Architecture,webdocuments,HTTP,Webservices.Uniform Resource Locators (URL) and Universal Resource Identifier (URI). Multimedia protocols-RTP, RTCP.

C. Text Books:

1. William Stallings, ― Data and Computer Communications, Eighth Edition, Pearson Prentice Hall, 2016, ISBN 0132433109.
2. Behrouz A. Fourouzan, ―Data Communications and Networking‖, Tata McGraw-Hill Education, 2006, ISBN 0070634149, 9780070634145.
3. Andrew S. Tanenbaum, ―Computer Networks‖, 4/e, Pearson education, 2003, ISBN 8131701980, 9788131701980

D. Reference Books:

1. James F. Kurose and Keith W. Ross, ―Computer Networking – A Top-Down Approach Featuring the Internet‖, 3/e, Pearson Education India, 2005, ISBN 8177588788, 9788177588781.
2. S. Keshav, ―An Engineering Approach To Computer Networking: ATM Networks, The Internet, And The Telephone Network‖, Pearson education, 2002, ISBN 8131711455, 9788131711453.
3. Halsall, Data Communication, ―Computer Networks and Open Systems‖, Pearson, 2003, ISBN 8178080982, 9788178080987.
4. W.R.Stevens, Kevin R. Fall, ―TCP/IP Illustated‖, Volume 1, 2/e, Addison-Wesley, ISBN 0132808188, 9780132808187.
5. Gary R. Wright, W. Richard Stevens , ―TCP/IP Illustrated‖, Volume 2, Addison-Wesley Professional, 1995, ISBN 0321617649, 9780321617644.
6. Douglas Comer, ―Internetworking with TCP/IP: Principles, protocols, and architecture‖ illustrated, Prentice Hall, 2006, ISBN 0131876716, 9780131876712.
7. Sam Halabi,―Internet Routing Architectures‖, Pearson Education India, 2008, ISBN 8131725944, 9788131725948.
8. Larry L. Peterson and Bruce S. Davie, ―Computer Networks: A System Approach‖,5, revised, Elsevier, 2011, ISBN 0123850606, 9780123850607.

Name of the Module: Software Engineering
Module Code: CS - 307
Credit Value: 3 {L=3, T=0, P=0}

A. Course Objectives:

The course is designed to meet with the objectives of understanding:

1. The need of software engineering, its different life cycles and different phases,
2. To measure cost, efforts, time and team management etc,
3. Testing and maintenance techniques of big projects and different risks and its management systems.

B. Course Content:

Overview of System Analysis & Design: Business System Concept, System Development Life Cycle, Waterfall Model, Spiral Model, Feasibility Analysis, Technical Feasibility, Cost- Benefit Analysis, COCOMO model.

System Requirement Specification: DFD, Data Dictionary, ER diagram, Process Organization & Interactions. System Design: Problem Partitioning, Top-Down And Bottop-Up design; Decision tree, decision table and structured English; Functional vs. Object- Oriented approach.

Coding & Documentation: Structured programming, OO programming, information hiding, Reuse, system documentation. Testing: Levels of Testing, Integration Testing, Test case Specification, Reliability Assessment, Validation & Verification Metrics, Monitoring & Control.

Software Project Management: Project Scheduling, Staffing, Software Configuration Management, Quality Assurance, Project Monitoring. Software modelling: with Unified Modelling Language. Case Tools: Concepts, use and application.

C. Text Books:

1. Roger S Pressman, ―Software Engineering – A practitioner‗s approach, McGraw-Hill HigherEducation, 2005, ISBN: 007301933X, 9780073019338
2. Rajiv Mall, ―Software Engineering, Prentice Hall of India. ISBN: 8120338197, 9788120338197
3. Carlo Ghezzi, Mehdi Jazayeri, Dino Mandrioli, ―Fundamentals of Software Engineering, Prentice Hall, 2003,2nd Edition, ISBN: 013099183X, 9780130991836

D. Reference Books:

1. Sommerville, ―Software Engineering, Pearson Education 2008, ISBN: 8131724611, 9788131724613
2. Behforooz, ―Software Engineering Fundamentals, Oxford University Press, ISBN: 0195105397, 9780195105391
3. Carlo Ghezzi, Mehdi Jazayeri, Dino Mandrioli, ―Software Engineering, Prentice Hall of India, 2 nd Edition, ISBN: 013099183X, 9780130991836
4. PankajJalote, ―An Integrated Approach to Software Engineering, Springer Science & Business Media, 1997, ISBN: 0387948996, 9780387948997.

5. Stephen R. Schach, —Object-oriented and Classical Software Engineering, Edition 8, illustrated McGraw-Hill, 2010, ISBN: 0071081712, 9780071081719
6. Bharat Bhushan Agarwal, Sumit Prakash Tayal, —Software Engineering, Laxmi Publications, 2009, ISBN: 8190855913, 9788190855914
7. ClaesWohlin, Per Runeson, Martin Höst, Magnus C. Ohlsson, BjörnRegnell, Anders Wesslén,
8. Experimentation in Software Engineering, Springer Science & Business Media, 2012, ISBN: 3642290442,

Name of the Module: Computer Graphics and Multimedia

Module Code: CS - 308

Credit Value: 3 {L=3, T=0, P=0}

A. Course Objectives:

The course is designed to meet with the objectives of:

1. Introducing graphical techniques such as modelling, representation, illumination, shadowing, rendering and texturing.
2. To learn two dimensional and three dimensional computer graphics with comprehend advanced software tools of computer graphics.
3. To gain knowledge about multimedia components and characteristics.

B. Course Content:

Introduction to Computer Graphics & Graphics Systems: Overview of computer graphics, representing pictures; color models; storage tube graphics display, Raster scan display, printers etc.; Active & Passive graphics devices; Computer graphics software.

Scan conversion: Points & lines, Line drawing algorithms; DDA algorithm, Bresenham‘s line algorithm, Circle generation algorithm; Ellipse generating algorithm; scan line polygon, fill algorithm, boundary fill algorithm, flood fill algorithm.

2D Transformation & Viewing: Basic transformations: translation, rotation, scaling; Matrix representations & homogeneous coordinates, transformations between coordinate systems; reflection shear; Transformation of points, lines, parallel lines, intersecting lines; clipping operations, point clipping, line clipping, clipping circles, polygons & ellipse.

3DTransformation&Viewing: 3Dtransformations: translation, rotation, scaling & other transformations. Rotation about an arbitrary axis in space, reflection through an arbitrary plane; general parallel projection transformation; clipping, viewport clipping, 3D viewing.

Curves: Curve representation, surfaces, designs, Bezier curves, B-spline curves.

Hidden Surfaces: Depth comparison, Z-buffer algorithm, Back face detection, BSP tree method, the Printer‘s algorithm, scan-line algorithm; Hidden line elimination.

Introduction to Multimedia: Concepts, uses of multimedia, text representation; Image, video and audio standards, compression techniques.

C. Text Books:

1. Hearn, Baker ―Computer Graphics (C version 2nd Ed.) – Pearson education.
2. Foley, Vandam, Feiner, Hughes –―Computer Graphics principles (2nd Ed.) – Pearson Education.
3. Z. Xiang, R. Plastock ― Schaum‘s outlines Computer Graphics (2nd Ed.) – Tata McGraw Hill

D. Reference Books:

1. F. Rogers, J. A. Adams ― Mathematical Elements for Computer Graphics (2nd Ed.) – Tata McGraw Hill
2. Mukherjee Arup, Introduction to Computer Graphics, Vikas Hill, Computer Graphics using open GL, Pearson Education
3. W. M. Newman, R. F. Sproull ―Principles of Interactive computer Graphics‖ – Tata McGraw Hill.
4. Computer Graphics and Multimedia: Applications, Problems and solution by John DiMarco, Idea Group Publication.
5. Computer Graphics, Multimedia and Animation by Malay K. Pakhira, Prentice-Hall
6. Multimedia, Computer Graphics and Broadcasting, by Taihoonkim, Hojjat Adeli

Name of the Module: Database Management System
Module Code: CS – 309
Credit Value: 3 {L=3, T=0, P=0}

B. Course Objectives:

The course is designed to meet with the objectives of

1. Understanding the purpose of a database management system (DBMS), the role of the database administrator
2. Understandingwhatismeantbydataconsistency,dataintegrity,dataredundancyanddata independence
3. Understanding the concept of entity relationships and data normalisation

C. Course Content:

Introduction: Concept & Overview of DBMS, Data Models, Database Languages, Database Administrator, Database Users, Three Schema architecture of DBMS.

Entity-Relationship Model: Basic concepts, Design Issues, Mapping Constraints, Keys, Entity-Relationship Diagram, Weak Entity Sets.

Relational Model: Structure of relational Databases, Relational Algebra, Relational Calculus, Extended Relational Algebra Operations, Views, Modifications of the Database.

SQL and Integrity Constraints: Concept of DDL, DML, DCL. Basic Structure, Set operations, Aggregate Functions, Null Values, Domain Constraints, Referential Integrity Constraints,assertions,views,NestedSubqueries,Databasesecurityapplicationdevelopment using SQL, Stored procedures and triggers.

Relational Database Design: Functional Dependency, Different anomalies in designing a Database.,Normalizationusingfunctionaldependencies,Decomposition,Boyce-CodeNormal Form, 3NF, Normalization using multi-valued dependencies.

Internals of RDBMS: Physical data structures, Query optimization, join algorithm, statistics and cost based optimization. Transaction processing, Concurrency control and Recovery Management, transaction model properties, state serializability, lock base protocols, two phase locking.

File Organization & Index Structures: File & Record Concept, Placing file records on Disk, Fixed and Variable sized Records, Types of Single-Level Index (primary, secondary, clustering), Multilevel Indexes, Dynamic Multilevel Indexes using B tree and B+ tree.

D. Text Books:

1. Henry F. Korth and Silberschatz Abraham, ―Database System Concepts,

Mc.GrawHil, 6th edition, ISBN: 0071289593,9780071289597.

2. ElmasriRamez and NovatheShamkant, —Fundamentals of Database Systems, Benjamin Cummings Publishing. Company, ISBN 8131716252,9788131716250.
3. Raghu Ramakrishnan, Johannes Gehrke, —Database Management System, McGraw-Hill, ISBN 0072465638,9780072465631.

E. Reference Books:

1. V.K. Jain, —Advanced Database Management System‖ ,Cyber Tech Publications, ISBN 8178840219,9788178840215.
2. DateC.J.,—AnIntroductiontoDatabaseSystems‖,PearsonEducationIndia,2006, ISBN8177585568,9788177585568.
3. Ullman JD., —Principles of Database Systems‖, Galgottia Publication, ISBN 8175155450,9788175155459.
4. JamesMartin,—PrinciplesofDatabaseManagement‖,PearsonEducation,ISBN 0137089171,9780137089178.
5. Arun K .Majumdar, Pritimay Bhattacharya, —Database Management Systems‖, TataMcGraw Hill, ISBN-13:978-0074622391.

Name of the Module: AI & Machine Learning Module
Code: CS - 310
Credit Value: 3{T=3, T=0, P=0}

A. Course Objectives:

- This introductory course gives an overview of many concepts, techniques, and algorithms in machine learning related to classification and regression problems.
- The course will give the student the basic ideas and intuition behind modern machine learning methods as well as a bit more formal understanding of how, why, and when they work.
- The underlying theme in the course is statistical inference as it provides the foundation for most of the methods covered.
- Make use of Data sets in implementing the machine learning algorithms.

B. Course Content:

Basics of Linear Algebra: Introduction to Machine Learning, linear classification, perceptron update rule, Perceptron convergence, generalization, Maximum margin classification, Classification errors, regularization.

Logistic regression, Linear regression, estimator bias and variance, active learning, Active learning, non-linear predictions, Kernal regression, kernel optimization, Model selection criteria, Description length, feature selection, expectation maximization

Classification problems; decision boundaries; nearest neighbor methods, Probability and classification, Naive Bayes, Bayes' Rule and Naive Bayes Model, Hidden Markov models (HMMs), Bayesian networks, Learning Bayesian networks, Logistic regression, online gradient descent, neural network, support vector machine (SVM), kernel ridge regression

Ensemble methods: Bagging, random forests, boosting, Unsupervised learning: clustering, k-means, hierarchical agglomeration, Advanced discussion on clustering, Latent space methods; PCA, Text representations; multinomial models; clustering and latent space models.

C. Text Books:

1. Machine Learning. Tom Mitchell. First Edition, McGraw-Hill.
2. Pattern Classification. Richard Duda, Peter Hart and David Stock. Second Edition, Wiley-Interscience.

D. Reference Books:

1. Simon Haykin, Neural Networks and Learning Machines Third Edition, Pearson Publisher
2. Christopher M. Bishop, Pattern Recognition and Machine Learning (Information Science and Statistics), Springer, 2006

Name of the Module: Networking
Laboratory Module Code: CS – 311
Credit Value: 1 {L=0, T=0, P=2}

List of Practical:

1. Study of different types of cross-wired cable and straight throughcable.
2. Study of Basic network commands and network configurationcommands.
3. Socket programming using Java or C programminglanguage.
4. Network topology configuration using Cisco packet tracersoftware
5. Network topology configuration of static routing using Cisco packet tracersoftware
6. Routing Protocol Configuration of a network using Cisco Packet Tracer Software (Eg. Static routing, RIP, RIP Version 2etc)
7. Firewall Configuration using ip tables and ip chains and solve different general problems in Linux OS.
8. Practical on Server Configuration Example, Web Server, Mail Server, FTP Server,DHCP, NFSetc.
9. Introduction to ns2 (network simulator) - small simulation exercises to study TCP behaviour under different scenarios and study link layer protocols such as Ethernet and 802.11 wireless LAN.
10. Experimental study of application protocols such as HTTP, FTP, SMTP, using network packet sniffers and analyzers such as Ethereal. Small exercises in socket programming in C/C++/Java.
11. Experiments with packet sniffers to study the TCP protocol. Using OS (netstat, etc) tools to understand TCP protocol FSM, retransmission timer behaviour, congestion controlbehaviour.

Reference Books:

1. *J Gosling, B Joy, G L Steele and G Bracha, The Java Language Specification, 2/e, Addison-Wesley, 2000.*
2. *B Stroustrup, The C++ Programming Language, 3/e, Addison-Wesley Longman ReadingMA,1997.*
3. *S B Lippman, C++ Primer, 2/e, Addison-Wesley,1991.*
4. *T Budd, C++ for Java Programmers, Addison Wesley,1999.*
5. *M C Daconta, Java for C/C++ programmers, John Wiley & Sons,1996.*

Name of the Module: Database Management System Laboratory
Module Code: CSE - 312
Credit Value: 1 {L=0, T=0, P=2}

List of Practicals:

1. ER modelling of real world problems
2. Creating relational databases with simple tables
3. Implementing DDL and DML commands with examples
4. Implementing key constraints
5. Implementing Aggregate functions
6. Implementing Joins
7. Creating views and queries based on views
8. Implementing embedded SQL queries

Text Book:

Mastering Database Technologies, Ivan Bayross, BPB Publication

Name of the Module: Computer Graphics and Multimedia Laboratory
Module Code: CSE - 313
Credit Value: 1 {L=0, T=0, P=2}

Laboratory experiments will be based on the materials covered in the theory of this paper, specially emphasize on the following topics.

1. Write a program to draw a line through Bresenham's algorithm.
2. Write a program to draw a line through DAA algorithm.
3. Write a program for line drawing using symmetric DAA algorithm.
4. Write a program to draw a line through mid-point algorithm.
5. Write a program to draw a circle using mid-point algorithm.
6. Write a program to draw an Ellipse using mid-point algorithm.
7. Write a program for boundary fill algorithm.
8. Write a program for Flood fill algorithm.
9. Write a C program to clip a line using Cohen-Sutherland clipping algorithm.
10. Write a code for implementing the Liang-Barsky line clipping algorithm.
11. Write a code for implementing the Sutherland polygon clipping algorithm.
12. To write a C program to perform 2D basic transformations such as translation, rotation, scaling, shearing and reflection on the user-defined polygon figure.
13. To write a code for drawing the Bezier curve for the user-defined control polygon using mouse.
14. To write a C program to perform 3D transformations such as translation, rotation, scaling, reflection and shearing on user-defined origin situated in a cube.
15. Write a code for simulating analog clock (simple animation).

Text Books:

1. *Hearn, Baker—Computer Graphics(Cversion2nd Ed.)‖– Pearson education.*
2. *Foley, Vandam,Feiner, Hughes ––—Computer Graphics principles (2nd Ed.) – PearsonEducation.*

Name of the Module: Machine Learning Laboratory
Module Code: CSE - 314
Credit Value: 1 {L=0, T=0, P=2}

Laboratory experiments will be based on the materials covered in the theory of this paper.

1. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
2. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.
3. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.
4. Build an Artificial Neural Network by implementing the Back propagation algorithm and test the same using appropriate data sets.
5. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Python ML library in the program.
6. Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Python ML library can be used for this problem.
7. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.

Mini Projects:

Lab 8: Face Recognition Using Eigen faces (PCA Algorithm)

Lab 9: Heart Disease Prediction Using support vector machine

Lab 10: Rainfall prediction using linear regression

Proposed Courses for Seventh Semester

7^{Th} Semester						
Sl. No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	OE-XXX	Open Elective	3	0	0	3
2	CS – 401	Internet and Web Technology	3	0	0	3
3	CS– 402	Cryptography & network Security	3	0	0	3
4	CS– 403X	Elective-I	3	0	0	3
5	CS– 404	Pattern Recognition & Image Processing	3	0	0	3
6	CS – 405	Internet and Web Technology laboratory	0	0	2	1
7	CS – 406	Cryptography & Network Security laboratory	0	0	2	1
8	CS – 407	Pattern Recognition & Image Processing laboratory	0	0	2	1
9	CS– 492	**Summer Internship-III**	0	0	0	1
Total Credit			**15**	**0**	**6**	**19**

Name of the Module: Internet and Web Technology
Module Code: CS – 401
Credit Value: 3 {L=3, T=0, P=0}

A. Course Objectives:

The course is designed to meet the objectives of:

a) To complete an in-depth knowledge of web technology,

b) Toknowandtohavetheideafordifferentwebapplicationthatmostwebdevelopersarelikelyto use.

c) To be aware of, and to have used, the enhancements of the web applications.

d) To know the different types of web application software.

B. Course Content:

Introduction to web technologies and architectures: Recent Web technologies: A case study on WWW, web2.0 etc., Client/Server Computing: C/S Computing, Middleware, Fat client VS Fat Servers, N-tiered Software Architecture.

Commonly used protocols Protocols: HTTP, FTP, SMTP, POP etc.

Mark-language Markup Languages: SGML, DTD Resources, HTML, CSS, XML, XSL, Query Languages for XML.

Basics of web-programming Programming: Client side scripting: JAVASCRIPT, AJAX. Server side programming in PHP, Overview of Java, JAVA Applet

Web Servers and Servlets: Tomcat web server, Introduction to Servelets: Lifecycle of a Serverlet, JSDK, The Servlet API, The javax.servelet Package, Reading Servelet parameters, and Reading Initialization parameters. The javax.servelet HTTP package, Handling Http Request & Responses, Using Cookies-Session Tracking, Security Issues,

Introduction to JSP: The Problem with Servelet. The Anatomy of a JSP Page, JSP Processing. JSP Application Design with MVC Setting Up and JSP Environment: Installing the Java Software Development Kit, Tomcat Server & Testing Tomcat

JSP Application Development, Database Access: Database Programming using JDBC, Studying Javax.sql.* package, Accessing a Database from a JSP Page, Application - Specific Database Actions, Deploying JAVA Beans in a JSP Page, Introduction to struts framework.

E-commerce applications E-business models, E-commerce and WWW, secure electronic payment protocols, e-commerce payment systems, web-based marketing, search engine and directory registration, e-commerce site designing tools etc.

C. Text Books:

1. Xavier C, "Web Technology & Design, New Age Publication.
2. Herbert Schidt, “Java 2 Complete Reference”, 7th Edition", McGraw- Hill Education.
3. "Java Server Programming, J2EE edition. (VOL I and VOL II) ", WROX publishers.

D. Reference Books:

1. Chris Bates, "Web Programming, building internet applications, 2nd edition", WILEYDreamtech
2. Patrick Naughton and Herbert Schildt, "The complete Reference Java 2 Fifth Edition
3. Hans Bergsten "Java Server Pages ", SPD O 'Reilly.
4. Dietel and Nieto, "Internet and World Wide Web - How to program ", PHI/Pearson Education Asia
5. JoclSklar, "Web Warrier guide to web design technologies", Cengage Learning, New Delhi
6. Ian Graham, "The XHTML 1.0 Web Developement Sourcebook", Wiley
7. Ian Graham and Liam Quin, "The XML Specification Guide ", Wiley
8. Ian Graham, John, "The HTML Stylesheet Sourcebook", Wiley and Sons
9. Jeffrey C. Jackson, “Web Technologies: A computer science perspective” , Pearson Education
10. W3schools tutorials, http://www.w3schools.com/

Name of the Module: *Cryptography & Network Security*

Module Code: CS - 402

Credit Value: 3 {L=3, T=0, P=0}

A. Course Objectives:

The course is designed to meet with the objectives of:

1. Understanding the state-of-the-art of cryptography and their security goals.
2. Examining and studying of different cryptographic methods for achieving confidentiality of date.
3. Examining and studying of different cryptographic methods for achieving integrity of date.
4. Understanding message authentication and hash function
5. Understanding the various protocol in network security.

B. Course Content:

Introduction of Information Security: Cryptography, Conventional Encryption, Traditional technique: Substitution cipher, Transmission cipher, Stream Cipher, Block Cipher, Roaster Machine.

Modern Symmetric Techniques, Mathematics of symmetric key cryptography, Cryptanalysis of classical ciphers, General Attacks, Secret and Private Key Cryptography, DES, Modes of operation of DES, Automatic Variable Key, Proof of DES, Merits and Demerits of DES, Quantification of Performance, TDES, Advanced Encryption Standard/AES, Comparison of Secret Key Systems, Modes of operation of AES Limitations of AES, Limitation of Secret or Private Key Crypto systems. Asymmetric key cryptography: Mathematics of Asymmetric key cryptography, Public Key Cryptography RSA Algorithm, Limitations of RSA Algorithm, Comparison of RSA and TRAP DOOR Public Key Crypto systems, Rabin Cryptosystem, El Gamal Cryptosystem, Elliptic Curve Cryptosystems.

Key management: Key Transport Protocols, Needham Schroeder Protocol, Key Agreement Protocol, Diffie -Hellman Protocol, Station to Station Protocol, Merkless Puzzle Technique of key agreement, Public Key Distribution, Message integrity and message authentication, Cryptography hash function, Digital Signature, Entity Authentication.

Networks security: Application Layer: PGP and S/MIME, Transport Layer: SSL and TLS, Network Layer: IPSec.

C. Text Books:

1. AtulKahate, Cryptography and Network Security, Tata McGraw-Hill Education,07-2008.
2. Behrouz A. Forouzan & D. Mukhopadhyay - Cryptography and Network Security, Tata McGraw-Hill Education, 3E,2015
3. William Stallings - Cryptography and Network Security, Principle and Practice, Pearson

Education India, 5E,2011

D. Reference Books:

1. MichaelErschloe,—InformationWarfare:HowtoSurviveCyberAttacks‖, Osborne/McGraw-Hill,2001.
2. Brian Hatch, James Lee and George Kurtz, —Hacking Exposed: Linux: Linux Security Secrets and Solutions‖, Osborne McGraw-Hill,1983.
3. Kenneth R. Van Wyk, Richard Forno, —Incident Response‖, O'Reilly,2001.
4. KevinMandia,ChrisProsise,—IncidentResponse:InvestigatingComputerCrime‖, Osborne/McGraw-Hill,2001.
5. Mike Schiffman, —Hacker's Challenge‖, McGraw Hill Professional,2002.
6. Julia Allen,—The CERT Guideto System and Network Security Practices‖ Addison-Wesley, 2001.
7. RichardE.Smith—Authentication:FromPasswordstoPublicKeys‖,Addison-Wesley,2002.
8. StuartMcClure,SaumilShah,ShreerajShah,—WebHacking:AttacksandDefense‖, Addison-Wesley Professional, 2003.
9. Mike Shema, Bradley C. Johnson, Keith J. Jones, —Anti Hacker Tool Kit: Key Security Tools and Configuration Techniques‖, San Val, Incorporated,2002.

Name of the Module: Pattern Recognition & Image Processing
Module Code: CS – 404
Credit Value: 3{L=3, T=0, P=0}

A. Course Objectives:

The course is designed to meet the objectives of:

1. To equip students with basic mathematical and statistical techniques commonly used in pattern recognition and image processing.
2. To introduce students to a variety of pattern recognition algorithms.
3. Introduce the student to analytical tools which are currently used in digital image processing as applied to image information for human viewing.
4. To apply knowledge of different image processing schemes for real time applications.

B. Course Content:

Basic Concepts: Pattern Recognition Systems, Fundamental Problems in pattern recognition systemdesign, Design concepts and Methodologies, Character recognition, Speech recognition, Finger print Recognition — Pattern Recognition Model.

Decision Functions: Linear Decision functions Distance functions. Minimum distance classification, clustering concepts, Cluster seeking algorithms, Maximum distance, K- means Algorithms.

Bayes classified: decision function For Bayes classifier Bayes Classifier for normal patterns. Trainable pattern classifiers deterministic approach, perception approach reward punishment concept.

Gradient approach: Gradient Descent algorithms LMSE Algorithms Multi category classification.

Trainable pattern classifiers: statistical approach stochastic approximation methods, Robbin Minro algorithms increment correction algorithms, LMSE algorithms. Syntactic patter recognition formulation syntax directed recognition picture descript.

Digital Image fundamentals: Representation, elements image transforms Fast Fourier transform, DCT and DWT.

Image enhancement: Spatial domain - frequency domain methods Histogram, Modification techniques, Image smoothing, image sharpening.

Image encoding: Fidelity criteria, Encoding process, Mapping Quantizer coder, Image Segmentation Masks, Point detection, Line Detection, Edge Detection.

C. Text Books:

1. Digital Image Processing by R.C. Gonzalez & R.E. Woods, Addison Wesley.
2. Pattern Recognition Principles J.T.TOU.R.C. Gonzalez, Addison Wesley.
3. Fundamentals of Digital Image Processing by A.K. Jain, PHI Pearson Education

D. Reference Books:

1. DaishengLuo, —Pattern Recognition and Image Processing, Horwood, 1998

2. Cornelius T. Leondes, —Image Processing and Pattern Recognition, Elsevier
3. Frank Y. Shih, Image Processing and Pattern Recognition: Fundamentals and Techniques, John Wiley & Sons
4. James C. Bezdek, James Keller, Raghu Krisnapuram, Nikhil R. Pal, Fuzzy Models and Algorithms for Pattern Recognition and Image Processing, Springer
5. Ashish Ghosh, Sankar K. Pal, —Soft Computing Approach to Pattern Recognition and Image Processing, World Scientific Publishing Co. Pte. Ltd.
6. Jun Shen, Patrick Shen-pei Wang, Tianxu Zhang, —Multispectral Image Processing and PatternnRecognition, World Scientific Publishing Co. Pte. Ltd.
7. Tzay Y. Young, —Handbook of pattern recognition and image processing‖, Academic Press, 1994.

Name of the Module: Internet and Web Technology
Laboratory Module Code: CS - 405
Credit Value: 1 {L=0, T=0, P=2}

A. Course Objectives:

a) Implement interactive web page(s) using HTML, CSS and JavaScript.
b) Design a responsive web site using HTML5 and CSS3.
c) Demonstrate Rich Internet Application.
d) Build Dynamic web site using server side Servlet, JSP Programming and Database connectivity.
e) Describe and differentiate different Web Extensions and Web Services.

B. List of practical's:

1. ExperimentalstudyofapplicationprotocolssuchasHTTP,FTP,SMTP,usingnetwork packet sniffersandanalyzerssuchasEthereal.SmallexercisesinsocketprogramminginC/C++/Java.
2. Experiments with packet sniffers to study the TCP protocol. Using OS (netstat, etc) tools to understandTCP protocol FSM, retransmission timer behaviour congestion controlbehavior.
3. Introduction to ns2 (network simulator) - small simulation exercises to study TCP behaviour under differentscenarios.
4. Setting up a small IP network - configure interfaces, IP addresses and routing protocols to set up a small IP network. Study dynamic behaviour using packetsniffers.
5. Experimentswithns2tostudybehaviour(especiallyperformanceof)linklayerprotocolssuch as Ethernet and 802.11 wirelessLAN.
6. Practical on Server Configuration Example, Web Server, Mail Server, FTP Serveretc.
7. Practice on Cisco Packet Tracersimulator.
8. Basic use of HTML tag, linking image table, frame, formdesign.
9. DHTML- inline styles, creating style sheets with the style element, linking external style sheet, positioning elements, user stylesheet.
10. Creating eventhandlerthat respondsto mouse andkeyboardevent: Onload, onmouseover, onmouse out, onfocus, onblur, onsubmit, onresult, onclick,onchange.
11. Structuring data with xml, xml parser, extensible style language (xsl); customising markuplanguage.
12. Configuring apache-tomcat server. 6. Building simple jsp: Declaring variables and methods in jsp, inserting java expression in jsp, processing request from user, generating a dynamic response for the user. Accessing database from jsp, inserting applet intojsp

E. Text Books:

1. Xavier C, "Web Technology & Design, New Age Publication.
2. Herbert Schidt, “Java 2 Complete Reference”, 7th Edition", McGraw- Hill Education.
3. "Java Server Programming, J2EE edition. (VOL I and VOL II) ", WROX publishers.

Name of the Module: Cryptography and Network Security
Laboratory Module Code: CS - 406
Credit Value: 1 {L=0, T=0, P=2}

Laboratory is mainly based on the materials taught i.e. development of code for Classical Cryptosystems, DES, AES, IDEA, RSA, MD5, SHA, DSA etc. and do experimentation. Mini projects may be given in small groups.

List of practical:

1. Finding GCD of two integer numbers: Euclidian and Extended Euclidian Algorithm, Finding the inverse: Additive and multiplicative
2. Traditional Symmetric Cipher techniques:
    a) Mono alphabetic Cipher: Additive, Multiplicative, Affine
    b) Poly alphabetic Cipher: Auto key cipher, Play fair cipher, Hill cipher, Vigenerecipher.
    c) Transposition cipher
3. Modern Symmetric Encryption: Data Encryption (DES), Advance Encryption Standard (AES).
4. Asymmetric Cipher: RSA, Elgamal, Rabin, Elliptic Curve Cryptosystem.
5. Digital Signatures: RSA digital signature scheme, Elgamal digital signature, etc.
6. Entity Authentication: Challenge response, Zero knowledge, etc
7. Key management: Diffie- Hellman,etc.

Name of the Module: Pattern Recognition & Image Processing Laboratory
Module Code: CS - 407
Credit Value: 1 {L=0, T=0, P=2}

List of practical:

Lab 1: Write a program for image enhancement

Lab2: Write a program for image compression

Lab3: Write a program for color image processing

Lab4: Write a program for image segmentation

Lab 5: Program for edge detection algorithm.

Lab 6: To fill the region of interest for the image

Mini Projects

Lab 7: Take a hand written document, perform preprocessing and try to segment into characters

Lab 8: Face Recognition Using Eigen faces (PCA Algorithm)

Lab 9: Heart Disease Prediction Using support vector machine

Lab 10: Rainfall prediction using linear regression

Eighth Semester

8th Semester						
Sl. No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	CS – 408	Elective-II(Swayam/ NPTEL)	3	0	0	3
2	CS– 409	Elective-III(Swayam/ NPTEL)	3	0	0	3
3	CS – 498	Grand Viva	0	0	4	2
4	CS–499	Project Phase-II & Dissertation	0	0	20	10
		Total Credits	**9**	**0**	**18**	**18**

List of Subjects for Elective – I						
Sl. No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	CS – 403A	Data Compression	3	0	0	3
2	CS – 403B	Software Project Management	3	0	0	3
3	CS – 403C	Real Time Operating Systems	3	0	0	3
4	CS – 403D	Parallel Algorithms	3	0	0	3
5	CS – 403E	Virtualization and Cloud Computing	3	0	0	3
6	CS – 403F	Soft Computing	3	0	0	3
7	CS – 403G	Natural Language Processing	3	0	0	3
8	CS – 403H	Data Mining & Warehousing	3	0	0	3
9	CS – 403I	Distributed Operating System	3	0	0	3

ELECTIVE – I

Name of the Module: Data Compression
Module Code: CS- 403A
Credit Value: 3{L=3, T=0, P=0}

A. **Course Objectives:**

The course is designed to meet the objectives of:

1. Introduce the fundamentals concepts of Data Compression,
2. Equip students with the knowledge and skills of coding Theory,
3. Explore the different paradigms of image and data compression.

B. **Course Content:**

Introduction: Compression Techniques: Loss less compression, Lossy Compression, Measures of performance, Modeling and coding, Mathematical Preliminaries for Lossless compression: A brief introduction to information theory, Models: Physical models, Probability models, Markov models, composite source model, Coding: uniquely decodable codes, Prefix codes.

Huffman coding: The Huffman coding algorithm: Minimum variance Huffman codes, Adaptive Huffman coding: Update procedure, Encoding procedure, Decoding procedure. Golomb codes, Rice codes, Tunstall codes, Applications of Hoffman coding: Loss less image compression, Text compression, Audio Compression.

Arithmetic Coding: Coding a sequence, Generating a binary code, Comparison of Binary and Huffman coding, Applications: Bi-level image compression-The JBIG standard, JBIG2, Image compression. Dictionary Techniques: Introduction, Static Dictionary: Diagram Coding, Adaptive Dictionary. The LZ77 Approach, The LZ78 Approach, Applications: File Compression-UNIX compress, Image Compression: The Graphics Interchange Format (GIF), Compression over Modems: V.42 bits, Predictive Coding: Prediction with Partial match (ppm): The basic algorithm, The ESCAPE SYMBOL, length of context, The Exclusion Principle, The Burrows-Wheeler Transform: Move-to-front coding, CALIC, JPEG-LS, Multi- resolution Approaches, Facsimile Encoding, Dynamic Markoy Compression.

Mathematical Preliminaries for Lossy: Coding Distortion criteria, Models, Scalar Quantization: The Quantization problem, Uniform Quantizer, Adaptive Quantization, Non uniform Quantization. **Vector Quantization:** Advantages of Vector Quantization over Scalar Quantization, TheLinde-Buzo-Gray Algorithm, Tree Structured Vector Quantizers. Structured Vector Quantizers.

C. **Text Books:**

1. Khalid Sayood, ―Introduction to Data Compression‖, Morgan Kaufmann Publishers
2. Ida MengyiPu, ―Fundamental Data Compression‖, Butterworth-Heinemann, 2005
3. Adam Drozdek, ―Elements of Data Compression‖, Brooks/Cole-Thomson Learning, 2002

D. Reference Books:

4. David Salomon, ―Variable-length Codes for Data Compression, Springer, 2007
5. Khalid Sayood ,―Lossless Compression Handbook, Academic Press, 2002
6. Source Wikipedia, LLC Books, ―Compression Algorithms: Lossless Compression Algorithms, Lossy Compression Algorithms, Huffman Coding, Lossless Data Compression, Jpeg, General Books, 2010
7. KamisettyRamam Rao, Pat Yip , ―The Transform and Data Compression Handbook, CRC Press,2010
8. S Sharma, ―Fundamentals of Data Compression, S K Kataria and Sons.
9. Peter D. Symes, ―Video compression: fundamental compression techniques and an overview of the JPEG and MPEG compression systems‖, McGraw-Hill, 1998

Name of the Module: Software Project Management
Module Code: CS – 403B
Credit Value: 3{L=3, T=0, P=0}

A. Course Objectives:

The course is designed to meet the objectives of:

1. Deliver successful software projects that support organization's strategicgoals,
2. Match organizational needs to the most effective software developmentmodel,
3. Plan and manage projects at each stage of the software development life cycle(SDLC),
4. Create project plans that address real-world managementchallenges,
5. Develop the skills for tracking and controlling softwaredeliverables.

B. Course Content:

Fundamentals: Conventional Software Management – Evolution of Software Economics – Improving Software Economics – Conventional versus Modern Software Project Management.

Software Management Process Framework: Lifecycle Phases – Artifacts of the Process – Model Based Software Architectures – Workflows of the Process – Checkpoints of the Process.

Software Management Disciplines: Iterative Process Planning – Organization and Responsibilities – Process Automation – Process Control and Process Instrumentation – Tailoring the Process.

Managed And Optimized Process: Data Gathering and Analysis – Principles of Data Gathering – Data Gathering Process – Software Measures – Data Analysis – Managing Software Quality – Defect Prevention. Case Studies: COCOMO Cost Estimation Model – Change Metrics – CCPDS– R.

C. Text Books:

1. Walker Royce ―Software Project Management A Unified Framework, Pearson Education,2004
2. Humphrey Watts, ―Managing the software process, Addison Wesley, 1989. (Unit IV)
3. Ramesh Gopalaswamy, ―Managing Global Projects, Tata McGraw Hill, 2001.
4. Bob Hughes, Mikecotterell, ―Software Project Management, 3rd Edition, Tata McGraw Hill, 2004.

D. Reference Books:

1. Robert T. Futrell, Donald F. Shafer, Linda Shafer , ―Quality Software Project Management, Prentice Hall Professional, 2002
2. Robert Bruce Kelsey, ―Software Project Management: Measures for Improving Performance,Management Concepts, 2006
3. S. A. KELKAR, ―SOFTWARE PROJECT MANAGEMENT: A CONCISE STUDY‖, PHI Learning Pvt. Ltd., 2012

Name of the Module: Real Time Operating Systems
Module Code: CS – 403C
Credit Value: 3{L=3, T=0, P=0}

A. Course Objectives:

The course is designed to meet the objectives of:

1. Participant with basics of real-time operating systems
2. Give the participant knowledge and skills necessary to develop software for embedded computer systems using a real-time operating system.

B. Course Content:

Introduction to Operating System: Computer Hardware Organization, BIOS and Boot Process, Multi-threading concepts, Processes, Threads, Scheduling.

Basics of real-time concepts: Terminology: RTOS concepts and definitions, real-time design issues, examples, Hardware Considerations: logic states, CPU, memory, I/O, Architectures, RTOS building blocks, Real-Time Kernel

Process Management: Concepts, scheduling, IPC, RPC, CPU Scheduling, scheduling criteria, scheduling algorithms
Threads: Multi-threading models, threading issues, thread libraries

Mutex: creating, deleting, prioritizing mutex, mutex internals

Inter-process communication: buffers, mailboxes, queues, semaphores, deadlock, priority inversion, Pipes

Memory Management: process stack management, run-time buffer size, swapping, overlays, block/page management, replacement algorithms, real-time garbage collection
Kernel Design Issues: structure, process states, data structures, inter-task communication mechanism, Linux Scheduling

C. Text Books:

1. Hermann Kopetz, ―Real-Time Systems: Design Principles for Distributed Embedded Applications, Springer Science & Business Media, 2011, ISBN: 144198237X, 9781441982377
2. Philips A. Laplante, ―Real-Time System Design and Analysis, John Wiley & Sons, 2004, ISBN: 0471648280, 9780471648284

3. Doug Abbott , ―Linux for Embedded and Real-Time Applications, Newnes, 2012, ISBN: 0123914337, 9780123914330
4. Albert M. K. Cheng, ―Real-Time Systems: Scheduling, Analysis, and Verification‖, John Wiley &
5. Sons, 2003, ISBN: 0471460842, 9780471460848

D. Reference Books:

1. Francis Cottet, Joëlle Delacroix, Claude Kaiser, ZoubirMammeri, ―Scheduling in Real-Time Systems, John Wiley & Sons, 2002, ISBN: 0470847662, 9780470847664.
2. Joseph, M, ―Real-Time Systems Specification, Verification and Analysis. Prentice Hall, 1996, ISBN 0-13-455297-0.
3. Krishna, C. M., Shin, K. G., ―Real-Time Systems, McGraw-Hill, 1997, ISBN 0-07-114243-6.
4. Labrosse, J. J., ―MicroC OS II: The Real Time Kernel, Newnes, 2nd ed., 2002, ISBN 978-1578201037.

Name of the Module: Parallel Algorithm
Module Code: CS – 403D
Credit Value: 3{L=3, T=0, P=0}

A. Course Objectives:

The course is designed to meet the objectives of:

1.To acquaint students with the basic concepts of parallel and distributed computing.

2.The course aims to the general principles of parallel and distributed algorithms and their time complexity.

3.To Study different aspects of Parallel Models

4.To Study different aspects of Interconnection Architecture

5.Analyse fundamental parallel algorithms from various application domains.

B. Course Content:

Sequential model, need of alternative model, parallel computational models such as PRAM, LMCC, Hypercube, Cube Connected Cycle, Butterfly, Perfect Shuffle Computers, Tree model, Pyramid model, Fully Connected model, PRAM-CREW, EREW models, simulation of one model from another one.

Performance Measures of Parallel Algorithms, speed-up and efficiency of PA, Cost optimality, An example of illustrate Cost-optimal algorithms- such as summation, Min/Max on various models.

Parallel Sorting Networks, Parallel Merging Algorithms on CREW/EREW/MCC/, Parallel Sorting Networks on CREW/EREW/MCC/, linear array

Parallel Searching Algorithm, Kth element, Kth element in X+Y on PRAM, Parallel Matrix Transportation and Multiplication Algorithm on PRAM, MCC, Vector-Matrix Multiplication, Solution of Linear Equation, Root finding. Graph Algorithms - Connected Graphs, search and traversal, Combinatorial Algorithms- Permutation, Combinations, Derrangements

C. Text Books:

1. M.J. Quinn, ―Designing Efficient Algorithms for Parallel Computer, McGrawHill.
2. R. Greenlaw, H.J. Hoover, W.L. Ruzzo, Limits to Parallel Computation: P-Completeness Theory, Oxford University Press, New York, 1995.
3. V. Kumar, A. Grama, A. Gupta, G. Karypis, ―Introduction to Parallel Computing, The Benjamin/Cummings Publishing Company, Redwood City, California, 1994.

D. Reference Books:

1. *T. Cormen, C. Leiserson, R. Rivest, ―Introduction to Algorithms, The MIT Press,* Cambridge, 1992.
2. S. G. Akl, ―The Design and Analysis of Parallel Algorithms, Prentice Hall, 1989.M. J. Quinn, ―Parallel Computing, McGraw Hill, 1994.
3. F.T. Leighton, ―Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes, Morgan Kaufmann Publishers, San Mateo, California, 1992.
4. D.P. Bovet, P. Crescenzi, ―Introduction to The Theory of Complexity, Prentice Hall, N.Y., 1994.

5. Selim G. Akl, ―Parallel Sorting Algorithms, Academic Press, 2014, ISBN: 148326808X, 9781483268088

Name of the Module: Virtualization and Cloud Computing
Module Code: CS – 403E
Credit Value: 3 {L=3, T=0, P=0}

A. Course Objectives:

The course is design to meet the objectives of:

The student will learn about the cloud environment, building software systems and components that scale to millions of users in modern internet, cloud concepts capabilities across the various cloud service models including Iaas, Paas, Saas, and developing cloud based software applications on top of cloud platforms

B. Course Content:

Introduction Definitions, Characteristics of cloud computing, Advantages and disadvantages of cloud computing, Cloud computing Vs Grid computing, Cloud computing Vs Distributed computing, Cloud computing Vs Cluster Computing.

Virtualization Basic concept– Hypervisor- Types of virtualization- hardware, operating system, server, storage- Features of virtualization- Advantages and disadvantages of different types of virtualization.

Cloud Architecture Types of deployment models-Private, Public , Hybrid, Community, Types of service models-Iaas, PaaS, SaaS.

Cloud storage architecture Evolution of storage technology, storage models, file systems and database, distributed file systems, general parallel file systems. Google file system, Apache Hadoop, BigTable, Megastore, Amazon Simple Storage Service(S3).

Cloud Security Cloud vulnerabilities-Threats to cloud confidentiality-VM cross attack, Malicious Sys Admin- Defense mechanism-Co residency detection, No Hype-Threats to cloud integrity- data loss/manipulation, dishonest computation- Defense Mechanism-Provable Data Possession (PDP), Proof of Retrievability, Dynamic PDP.

C. Text Books:

1. Distributed and Cloud Computing, Kai Hwang, Geoffry C. Fox, Jack J. Dongarra MK Elsevier.
2. Cloud Computing, Theory and Practice, Dan C Marinescu, MK Elsevier.
3. Cloud Computing, A Hands on approach, ArshadeepBahga, Vijay Madisetti, University Press

C. Reference Books:

1. Cloud Computing, A Practical Approach, Anthony T Velte, Toby J Velte, Robert Elsenpeter, TMH

2. Mastering Cloud Computing, Foundations and Application Programming, Raj Kumar Buyya, Christen vecctiola, S Tammaraiselvi, TMH

Name of the Module: Soft Computing
Module Code: CS – 403F
Credit Value: 3 {L=3, T=0, P=0}

A. Course Objectives:

The student should be made to:

1. Learn the various soft computing frameworks
2. Be familiar with design of various neural networks
3. Be exposed to fuzzy logic
4. Learn genetic programming.

B. Course Content:

INTRODUCTION: Artificial neural network: Introduction, characteristics- learning methods – taxonomy – Evolution of neural networks- basic models – important technologies – applications. Fuzzy logic: Introduction – crisp sets- fuzzy sets – crisp relations and fuzzy relations: cartesian product of relation – classical relation, fuzzy relations, tolerance and equivalence relations, non-iterative fuzzy sets. Genetic algorithm- Introduction – biological background – traditional optimization and search techniques – Genetic basic concepts.

NEURAL NETWORKS: McCulloch-Pitts neuron – linear separability – hebb network – supervised learning network: perceptron networks – adaptive linear neuron, multiple adaptive linear neuron, BPN, RBF, TDNN- associative memory network: auto-associative memory network, hetero-associative memory network, BAM, hopfieldnetworks,iterativeautoassociativememorynetwork&iterativeassociativememorynetwork– unsupervised learning networks: Kohonenself organizing feature maps, LVQ – CP networks, ART network.

FUZZY LOGIC: Membership functions: features, fuzzification, methods of membership value assignments- Defuzzification: lambda cuts – methods – fuzzy arithmetic and fuzzy measures: fuzzy arithmetic – extension principle – fuzzy measures – measures of fuzziness -fuzzy integrals – fuzzy rule base and approximate reasoning : truth values and tables, fuzzy propositions, formation of rules-decomposition of rules, aggregation of fuzzy rules, fuzzy reasoning-fuzzy inference systems-overview of fuzzy expert system-fuzzy decision making.

GENETIC ALGORITHM: Genetic algorithm and search space – general genetic algorithm – operators – Generational cycle – stoppingcondition–constraints–classification–geneticprogramming–multileveloptimization–real life problem- advances in GA

HYBRID SOFT COMPUTING TECHNIQUES & APPLICATIONS: Neuro-fuzzy hybrid systems – genetic neuro hybrid systems – genetic fuzzy hybrid and fuzzy genetic hybrid systems–simplified fuzzy ART MAP–Applications: A fusion approach of multispectral images with SAR, optimization of traveling salesman problem using genetic algorithm approach, soft computing based hybrid fuzzy controllers.

C. Text Books:

1. S.N. Sivanandam and S.N.Deepa, “Principles of Soft Computing”, Wiley India Pvt Ltd, 2011.
2. J.S.R.Jang, C.T. Sun and E.Mizutani, “Neuro-Fuzzy and Soft Computing”, PHI / Pearson Education 2004.
3. S. Rajasekaran and G.A. Vijayalakshmi Pai, “Neural Networks, Fuzzy Logic and Genetic Algorithm: Synthesis & Applications”, Prentice-Hall of India Pvt. Ltd., 2006.

D. Reference Books:

1. George J. Klir, Ute St. Clair, Bo Yuan, “Fuzzy Set Theory: Foundations and Applications” Prentice Hall, 1997.
2. David E. Goldberg, “Genetic Algorithm in Search Optimization and Machine Learning” Pearson Education India, 2013.
3. James A. Freeman, David M. Skapura, “Neural Networks Algorithms, Applications, and Programming Techniques, Pearson Education India, 1991.

Name of the Module: Natural Language Processing

Module Code: CS – 403G

Credit Value: 3{L=3, T=0, P=0}

A. Course Objectives:

The course is designed to meet the objectives of:

1. To provide a general introduction including the use of state automata for language processing,
2. To provide the fundamentals of syntax including a basic parse,
3. To explain advanced feature like feature structures and realistic parsing methodologies,
4. To explain basic concepts of remotes processing,
5. To give details about a typical natural language processing applications.

B. Course Content:

Introduction: Introduction: Knowledge in speech and language processing – Ambiguity – Models and Algorithms – Language, Thought and Understanding. Regular Expressions and automata: Regular expressions – Finite-State automata. Morphology and Finite-State Transducers: Survey of English morphology – Finite-State Morphological parsing – Combining FST lexicon and rules – Lexicon-Free FSTs: The porter stammer – Human morphological processing

Syntax : Word classes and part-of-speech tagging: English word classes – Tagsets for English – Part-of-speech tagging – Rule-based part-of-speech tagging – Stochastic part-of-speech tagging – Transformation-based tagging – Other issues. Context-Free Grammars for English: Constituency –Context-Free rules and trees – Sentence-level constructions – The noun phrase – Coordination – Agreement – The verb phase and sub categorization – Auxiliaries – Spoken language syntax – Grammars equivalence and normal form – Finite-State and Context-Free grammars – Grammars and human processing. Parsing with Context-Free Grammars: Parsing as search A Basic Top-Down parser – Problems with the basic Top-Down parser – The early algorithm – Finite-State parsing methods.

Advanced Features And Syntax: Features and Unification: Feature structures – Unification of feature structures – Features structures in the grammar – Implementing unification – Parsing with unification constraints – Types and Inheritance exicalized and Probabilistic Parsing: Probabilistic context-free grammar – problems with PCFGs – Probabilistic lexicalized CFGs – Dependency Grammars – Human parsing.

Semantic: Representing Meaning: Computational desiderata for representations – Meaning structure of language – First order predicate calculus – Some linguistically relevant concepts – Related representational approaches – Alternative approaches to meaning. Semantic Analysis: Syntax-Driven semantic analysis – Attachments for a fragment of English – Integrating semantic analysis into the early parser – Idioms and compositionality – Robust semantic analysis. Lexical semantics: relational among lexemes and their senses – WordNet: A database of lexical relations The Internal structure of words – Creativity and the lexicon.

Applications: Word Sense Disambiguation and Information Retrieval: Selectional restriction-based disambiguation – Robust word sense disambiguation – Information retrieval – other information retrieval tasks. Natural Language Generation: Introduction to language generation – Architecture for generation – Surface realization – Discourse planning – Other issues. Machine Translation: Language similarities and differences – The transfer metaphor – The interlingua idea:

Using meaning – Direct translation – Using statistical techniques – Usability and system development.

C. Text Books:

1. Daniel Jurafsky& James H.Martin, — Speech and Language Processing, Pearson Education (Singapore) Pte. Ltd., 2002.
2. James Allen, —Natural Language Understanding, Pearson Education, 2003.
3. HinrichSchütze, —Foundations of Statistical Natural Language Processing, MIT press
4. Steven Bird, Ewan Klein, Edward Loper, —Natural Language Processing with Python, O‘ Reilly

D. Reference Books:

1. Ela Kumar, —Natural Language Processing, I. K. International
2. Philip M. McCarthy, ChutimaBoonthum-Denecke, ChutimaBoonthum, —Applied Natural Language Processing and Content Analysis, information science reference
3. NitinIndurkhya, Fred J. Damerau, —Handbook of Natural Language Processing, Second Edition, CRC Press
4. Robert Dale, Hermann Moisl, Harold Somers, —Handbook of Natural Language Processing.
5. Karen Sparck Jones, Julia R. Galliers, —Evaluating Natural Language Processing Systems: An Analysis and Review, Springer Science & Business Media, 1995, ISBN: 3540613099, 9783540613091
6. Bernadette Sharp, Rodolfo Delmonte, —Natural Language Processing and Cognitive Science: Proceedings 2014, Walter de Gruyter GmbH & Co KG, 2015, ISBN: 1501501313, 9781501501319

Name of the Module: Data Mining & Warehousing

Module Code: CS- 403H

Credit Value: 3{L=3, T=0, P=0}

A. Course Objectives:

The course is designed to meet the objectives of:

1. understand classical models and algorithms in data warehousing and data mining,

2. enable students to analyse the data, identify the problems, and choose the relevant models and algorithms to apply,

3. Assess the strengths and weaknesses of various methods and algorithms and to analyse their behaviour.

B. Course Content:

Data Warehousing: Data warehousing Components Building a Data warehouse Mapping the Data Warehouse to a Multiprocessor Architecture DBMS Schemas for Decision Support Data Extraction, Cleanup, and Transformation Tools Metadata.

Business Analysis: Reporting and Query tools and Applications Tool Categories The Need for Applications Cognos Impromptu Online Analytical Processing (OLAP) Need Multidimensional Data Model OLAP Guidelines Multidimensional versus Multi relational OLAP Categories of Tools OLAP Tools and the Internet.

Data Mining: Introduction Data Types of Data, Data Mining Functionalities Interestingness of Patterns, Classification of Data Mining Systems, Data Mining Task Primitives, Integration of a Data Mining System with a Data Warehouse, Issues Data Preprocessing.

Association Rule Mining and Classification: Mining Frequent Patterns, Associations and Correlations, Mining Methods Mining Various Kinds of Association Rules Correlation Analysis Constraint Based Association Mining Classification and Prediction Basic Concepts Decision Tree Induction Bayesian Classification Rule Based Classification Classification by Backpropagation Support Vector Machines Associative Classification Lazy Learners Other Classification Methods and Prediction

Clustering and Applications and Trends in Data Mining: Cluster Analysis, Types of Data Categorization of Major Clustering Methods Kmeans Partitioning Methods Hierarchical Methods Density Based Methods Grid Based Methods Model-Based Clustering Methods Clustering High Dimensional Data Constraint Based Cluster Analysis Outlier Analysis Data Mining Applications.

C. Text Books:

1. *Data Warehouse Design: Modern Principles and Methodologies, McGrawHill, 2009 by M. Golfarelli, S. Rizzi. ISBN:978-0-07-067752-4.*
2. *Data Warehousing Fundamentals, John Wiley & Sons, 2006, by Paulraj Ponniah. ISBN:0-471-41254-6.*
3. *The Data Warehouse Lifecycle Toolkit, John Wiley & Sons, 2nd Edition, 2008, by Ralph Kimball, Margy Ross, Warren Thornthwaite, Joy Mundy, Bob Becker. ISBN: 978-1-118-07956-0.*
4. *The Data Warehouse ETL Toolkit, John Wiley & Sons, 1st Edition, by Ralph Kimball & Joe Caserta. ISBN: 978-0-7645-6757-5.*

5. *AlexBersonandStephenJ.Smith,DataWarehousing,DataMining&OLAP,TataMcGraw Hill Edition, Tenth Reprint 2007.*
6. *Jiawei Han and MichelineKamber, —Data Mining Concepts and Techniques, Third Edition, Elsevier,2007.*
7. *Pang-NingTan,MichaelSteinbachandVipinKumar,IntroductionToDataMining,Person Education,2007.*
8. *K.P.Soman,ShyamDiwakarandV.Ajay,InsightintoDataminingTheoryandPractice, Easter Economy Edition, Prentice Hall of India,2006.*
9. *G.K.Gupta,IntroductiontoDataMiningwithCaseStudies,EasterEconomyEdition, Prentice Hall of India,2006.*
10. *Daniel T.Larose, Data Mining Methods and Models, Wile-Interscience,2006.*
11. *Margaret H. Dunham, Data Mining: Introductory and Advanced Topics, Prentice Hall,2003.*
12. *Arun K Pujari, —Data Mining Techniques, Second Edition, UniversityPress.*

D. Reference Books:

1. *The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling, 3rd Edition, John Wiley & Sons, 2013, by Ralph Kimball &Margy Ross. ISBN: 978-1-118-73228-1*
2. *The Data Webhouse Toolkit: Building the Web-Enabled Data Warehouse, John Wiley & Sons, 1st Edition, by Ralph Kimball & Richard Merz. ISBN: 978-0-471-37680-4.*

Name of the Module: Distributed Operating System
Module Code: CS – 403I
Credit Value: 3 {L=3, T=0, P=0}

A. Course Objectives:

The course is design to meet the objectives of:

1. Study of basic structure of Operating System.
2. Difference between OS and Distributed OperatingSystems.
3. Study of Memory Management, Synchronization, Fault Tolerance,Deadlock.
4. Study of Multiprocessor Operating System, its Architecture andManagement.

B. Course Content:

Introduction: Function of an operating system, Design approaches, concepts of processes, threats, Critical Section problem, Other synchronization problem, Communicating sequential processes. **Process Deadlocks:** Introduction, Preliminaries, Models of Deadlocks, Models of resources, A graph theoretic models of System State, Necessary and sufficient conditions for a Deadlocks. Detection, Prevention and Avoidance.
Distributed Operating Systems: Introduction, System Architectures, issues in Distributed Operating Systems, Communication Networks, and Limitations of Distributed OS.

Distributed Mutual Exclusion: Classifications of Mutual Exclusion, Preliminaries, Solutions of Mutual Exclusions- Non token based algorithm, Lamport's Algorithm, Maekawa's Algorithm, Token Based Algorithm etc.

Distributed Deadlock Detection: Introduction, Preliminaries, Deadlock handing strategies in DistributedOS,ControlorganizationsforDistributedDeadlockDetection,CentralDeadlockDetection Algorithms, Distributed Deadlock DetectionAlgorithm.

AgreementProtocol:Systemmodel,Classificationofagreementproblems,ApplicationofAgreement Protocol.

Distributed File Systems: Mechanisms for building Distributed File Systems, Design Issues, Case Studies- Sun Network File System, The Sprite File Systems, Apollo Domain Distributed File System, Coda, Log Structure File Systems, and Disk Space Management.
Distributed Shared Memory: Architecture and Motivation, Algorithm for Implementing of DMS, Memory Coherence, Coherence Protocol, Case Studies- IVY, Mirage, Clouds.

Distributed Scheduling: Issues in load Distributing, Components of Load Distributing Algorithm, Stability, Load Distributing Algorithm, Performance Comparison, Selecting a Suitable Load Sharing Algorithm, Load Sharing Policies.

Failure Recovery: Classification of Failures, Backward and Forward Error recovery, Recovery in concurrent systems, Check points.
Fault Tolerance: Atomic Action and Committing, Commit Protocols, Non-Blocking Commit Protocols, Voting Protocols, Dynamic Voting Protocols, Case studies-Fault Tolerance under UNIX. **Multiprocessor System Architecture:** Motivations, Basic Architecture, Caching, Hypercube Architecture,
Multiprocessor Operating Systems: Structure, Multiprocessor Design issues, Threats, Process, Synchronization, Processor Scheduling, Memory Management, Reliability/Fault Tolerance

C. Text Books:

1. Singhal, “Advanced Concepts In Operating Systems”, Tata McGraw-Hill Education, 2001.
2. W. Richard Stevens, Stephen A. Rago, “Advanced Programming in the UNIX Environment, Addison-Wesley Professional Computing Series”, Addison-Wesley, 2013.
3. Andrew S. Tanenbaum, “Distributed Operating Systems", Pearson Education India, 1995.
4. Wiseman, Yair, “Advanced Operating Systems and Kernel Applications: Techniques and Technologies: Techniques and Technologies”, IGI Global, 2009

D. Reference Books:

1. Pramod Chandra P. Bhatt, “An Introduction to Operating Systems: Concepts and Practice”, PHI Learning Pvt. Ltd., 2010.
2. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, “Operating System Concepts, 9th Edition”, Addison-Wesley, 2013.
3. SibsankarHaldar, Alex AlagarsamyAravind, “Operating Systems”, Pearson Education India, 2010.
4. Chow, “Distributed Operating Systems And Algorithm Analysis”, Pearson Education India, 2009
5. Pradeep K. Sinha, “Distributed Operating Systems: Concepts And Design”, Phi Learning Pvt. Ltd., 1998.
6. Thomas W. Doeppner, “Operating Systems In Depth: Design And Programming”, John Wiley & Sons, 2011.