

॥ अंतरी पेटवू ज्ञानज्योत ॥  
उत्तर महाराष्ट्र विद्यापीठ, जळगांव

जा. क्र. उमाधि/१२/सिले/५५५३ /२७-२८

दिनांक: २९.८.१९९७

प्रति,

१. मा. परीक्षा निदेशक, उत्तर महाराष्ट्र विद्यापीठ, जळगांव.

२. मा. उपकुलसचिव, परीक्षा- पूर्व, उ. म. वि. जळगांव.

३. मा. सहा. कुलसचिव, परीक्षा - उत्तरार्ध, उ. म. वि. जळगांव.

४. आ. प्रोग्रामर, संगणक विभाग, उ. म. वि. जळगांव.

५. मा. विभाग प्रमुख

विभाग, उ. म. वि. जळगांव.

विषय :- अभ्यासक्रम पाठविणेबाबत.  
महोदय,

उपरोक्त विषयाच्या अनुमाने विदित करण्यात येते की, उत्तर महाराष्ट्र विद्यापीठाच्या संगणक विभागात गुरु अक्षलेल्या विविध कोर्सेस- साठी जून, १९९७ पासून नवीन सुधारीत अभ्यासक्रम लागू करण्यात आलेले आहेत. त्यासाठी खालील कोर्सेसचे नवीन सुधारीत अभ्यासक्रमाच्या प्रत्येकी एक-एक प्रती सोबत आपल्या माहितीसाठी पाठविलेल्या आहेत.

कळावे ही विनंती.

आपला विश्वासू,

उपकुलसचिव

- 1) M.Sc - Computer.
- 2) M.C.A.
- 3) B.B.S.
- 4) M.Sc.- Pesticides & Agrochemicals
- 5) M.Sc.- Polymer Chemistry
- 6) M.Sc.- Industrial Chemistry
- 7) M.Sc.- Statistics
- 8) M.Sc.- Computational Mathematics.

Spp\*

कळावे ही विनंती

PRICE Rs.10/-

**NORTH MAHARASHTRA UNIVERSITY**

**JALGAON - 425 001**

**Syllabus**

**FOR M.Sc. ( Computer Science )**

**with effect from July 1997**

**DEPARTMENT OF COMPUTER SCIENCE**

## STRUCTURE OF M.Sc. (Comp. Sci.) IMPLEMENTED FROM 1997-98

Course Code	Course Name	Lecture	Practical
<b><u>SEMESTER-I</u></b>			
CS-101	Computer programming and problem solving in C/C++	04	---
CS-102	Advanced database management concepts	04	---
CS-103	Modern operating systems	04	---
CS-104	Design and Analysis of Algorithms	04	---
CS-105	Computer Architecture	04	--
CS-LAB-I	RDBMS, C/C++ Programming	---	04
CS-LAB-II	UNIX, WINDOWS-NT, Design and analysis of algorithms	---	04
<b><u>SEMESTER-II</u></b>			
CS-201	Computer Graphics	04	---
CS-202	Parallel Computing	04	---
CS-203	Artificial Intelligence and Expert systems design.	04	---
CS-204	Software Engineering	04	---
CS-205	Theoretical Computer Science	04	---
CS-LAB-III	Parallel Programming, A.I. and Expert system design	---	04
CS-LAB-IV	Computer Graphics, Case Tools (Turbo Analyst, ERWIN)	---	04
<b><u>SEMESTER-III</u></b>			
CS-301	Compiler Construction	04	---
CS-302	Computer Networks and Distributed Computing	04	---
CS-303	Optimization Techniques	04	---
CS-304	Principles of Programming languages.	04	---
CS-305	Current Computing Trends	04	---
CS-LAB-V	Visual Basic, Windows Programming, Optimization Techniques.	---	04
CS-LAB-VI	Compiler Construction, Visual C++ & Network programming.	---	04
<b><u>SEMESTER-IV</u></b>			
Full time SIX months Industrial Training.			

## CS 101 COMPUTER PROGRAMMING AND PROBLEM SOLVING IN C/C++

**C language programming** : Data types, expression evaluation, precedence rules, type conversion, sequential structure, selective structure, repetitive structure( including recursion), array, pointers,, structure and unions, operation on bits, file processing, preprocessor

**Object Oriented Concepts** : objects, classes, messages, inheritance, dynamic binding, polymorphism, OO paradigm, reusability and extensibility, abstract data types, encapsulation, information hiding, genericity.

**Data abstraction in C++** : classes, ADT implementations, interfaces and implementation, member, methods, static members, member functions, public and private, initialization, constructors and destructors, operators and functions overloading, io operators, parameter passing by value and reference, function returning a reference, function signatures and name mangling, dynamic memory allocation and new and delete, assignment operator overloading, shallow and deep copies, copy constructor, encapsulation and friend functions and classes , genericity and template functions and classes

Container classes , iterators, istream class library, error handling and exceptions in c++

**Inheritance in c++** : base and derived classes . public, private and protected derivations, control of access and visibility using public/private, protected keywords , type compatibility among super and sub-types value pointer/reference assignment semantics, virtual functions and polymorphism, multiple inheritance, virtual derived classes Object oriented design and programming using classes and inheritance.

### Relevant books:

1. The C++ programming language by Bjarne Stroustrup Addison Wesley.
2. Object oriented programming with C++ by W. Balgurusamy, TMH.
- 3 An introduction to object oriented programming by Timothy Budd , Addison Wesley
4. C programming by Kernighan and Ritchie.

## CS 102 ADVANCED TOPICS IN DATABASE THEORY AND APPLICATION

**Distributed databases** : Introduction to distributed databases, networks, data distribution , object naming , distributed query processing , consistency , concurrency control , distributed commitment and recovery , deadlocks in distributed systems , security and protections, homogenous and heterogeneous system.

**Object databases** : concept of object , Introduction to object approach and object databases.

Concept of ODBC(object database connectivity)

### ORACLE :

**SQL\*PLUS** : designing reports using SQL\*PLUS

**PL/SQL** : Procedural language constructs in SQL , cursor movement , iterative control, conditional control.

**SQL\*Report Writer** . Menu driven report writing tool, concepts of report writer objects, Process involved in various types of reports , sample master-detail and cross-tab reports.

**Utilities** . Backup and Recovery using export/import , loading flat file data into oracle via SQL\* Loader

**SQL\*Forms 3.0** : The oracle 4GL , menu driven application building tools , various form characteristics and feature . concepts of application development , master detail relationships and pop-up windows.

**Pro C** : Elements of a Pro C program , defining queries, committing and rolling back work, error detection and recovery , dynamically defined statement

**Introduction to DBA** Oracle RDBMS concepts, file structure, table spaces and segments, user database concept, data dictionary , memory structure, process structure, SQL statement processing, transactional control.

### Relevant Books :

1. Database System Concepts by Korth and Stiberschutz. TMH
2. ORACLE 7.1 : Installation and User manual.
3. Developer 2000 : Installation and User manual.

## CS 103 MODERN OPERATING SYSTEM

**Introduction to Distributed system:** Goals, hardware concepts, software concepts, design issues.

**Communication in Distributed systems :** Layered protocols, the clients server model, remote procedure call, group communication.

**Synchronization in Distributed systems :** Clock synchronization, mutual exclusion, election algorithms, automatic transactions, deadlocks in distributed systems.

**Processes and processors in distributed systems :** threads, system models, processor allocation, scheduling in distributed systems.

**Distributed file systems:** Distributed file system design ,distributed file system implementation, trends in distributed file systems.

**Case study :** Detail study of modern OS :- MS windows-NT, Comparative study of Novell Netware and MS windows NT

### Relevant Books:

1. Modern operating systems A. S. Tanenbaum , PHI
2. Novell netware 4.1 manuals
3. MS Windows -NT manuals

## CS 104 DESIGN AND ANALYSIS OF ALGORITHMS

**Sorting Algorithms;** String processing, Graph and Geometric Algorithms; Most exploited paradigms for problem solving; Analysis of algorithms and computation of upper bound of complexity, Techniques of computation of lower bounds for the complexity of algorithms; Definition of non deterministic polynomial algorithms and the proof establishing NP-completeness of some problems.

### Relevant Books :

1. Horowitz and Sahani : Computer Algorithms. Galgotia Pub. India
2. D.E. Knuth : The art of programming VOL III. Addison Wesley Inc.
3. Aho, Hopcraft and Ullman : Data Structures and algorithms.

## CS -105 COMPUTER ARCHITECTURE

**Introduction of Parallel Processing :** Parallel processing mechanism, Parallelism in uniprocessor system.

**Parallel Computer Structures :** Architecture classification scheme.

**Pipelining and vector processing :** Instruction and arithmetic pipelines, vector processing requirements, pipeline computers and vectorization methods, various vector processors- STAR 100, Cray, CYBER-205, Fujitsu-200 and their special features.

**SMID Array processors :** Parallel algorithms for array processors, SMID computers and performance enhancement.

**Multiprocessor architecture and Programming :** functional structures, Interconnection networks, Parallel memory organizations, Multiprocessor Scheduling strategies, Parallel algorithms for Multiprocessor-synchronous and asynchronous.

**Data flow computers :** data driven computing and languages : Advantages and Potential difficulties. Etc.

### Relevant books :

1. Hawang K & Briggs F A: Computer Architecture and Parallel Processing, McGraw Hill
2. Koggi H: The Architecture of Pipelined Computers, McGraw Hill
3. Baer J L : Computer system Architecture, Computer Science Press

## CS LAB I : RDBMS, C/C++ PROGRAMMING

### RDBMS ( ORACLE )

1. Designing a database & tables and manipulation of data with SQL\*FORMS
2. querying the database with SQL and SQL\*PLUS
3. defining, using and querying one table views
4. defining, using and querying multiple table views with JOIN and without JOIN
5. data manipulation through views
6. working in detail with SQL\*Report writer
7. working in detail with SQL\*Forms
8. working in detail with SQL\*Menu
9. programming with ORACLE using PROC\*C
10. working with indexes and clusters

### C/C++ PROGRAMMING

#### C PROGRAMMING

1. (a) Write a program in C to solve  
$$S=(P*R*T)/100$$
  
(b) Write a program in C to exchange the value of two variables
2. (a) Write a program in C to find out the sum of the digits of a numbers  
(b) Write a C program to arrange N numbers in ascending order.
3. (a) Write a program in C to find out the reverse the digits of a number  
(b) Write a program to print Fahrenheit temperature their centigrade in Celsius equivalently using the formulae  
$$C=(5/9)(F-32)$$
4. (a) Write a C program to calculate the real roots of the quadratic equation  
$$AX^2+BX+c = 0$$
  
(b) Write a program in C to evaluate the formula  
$$Y=U+U/2+U/3+U/4$$
  
where  $U=(X-1)/X$
5. (a) Write a program in C to calculate the sum of the first 100 numbers of the following series  
2,4,6,8,.....  
(b) Write a program in C to generate the first 50 numbers of the following series  
0,1,1,2,3,5,8, ....
6. (a) Design an algorithm to evaluate the function  $\sin(x)$  as defined by the infinite series expansion  
$$\sin(x) = x/1! - x^3/3! + x^5/5! - \dots$$
  
(b) Write a program to find sum of the COS series.
7. (a) Write a C program to input any 20 numbers and find out the lowest one.  
(b) Write a C program to input any 20 numbers and find out the greatest one.
8. (a) Write a C program to find out the first 100 prime numbers.  
(B) Write a C program to find out the leap year with a suitable message.
9. Write a C program to input any 20 name and print only those names which are started by S.
10. Write a program to store 10 names in one dimensional array. Sort them alphabetically and print the sorted array.
11. Write a program to check whether inputted word is palindrome.
12. Write a program to  
(a) Concatenate two strings  
(b) Count the words in inputted sentence
13. Write a program to store the names of ten students and marks obtained in three subjects by each student in two dimensional array using pointers; sort the above array. Display the merit list.
14. Write a program to create a file (use structure) with following structure:  
Name     25 characters  
Gross Pay 5 digits

Read the above file and print the payroll with appropriate heading.  
15. Write a general program to convert inputted amount in words. For example after inputting 234.60 the result should be Rs. Two hundred thirty four

### C++ PROGRAMMING -

1. Write a c++ program that tells you how many bits are used to represent objects of type char, short, int, and long.
2. Write a hash table class that stores and retrieves records with a character string key. Provide public member function to insert, lookup, and remove records from hash table. Hide the implementation details in the private part of the class.
3. Design a directed graph data type class.
4. Design a two - way list whose implementation uses only a single link pointer. Include in your design an iterator that can traverse the list in either direction.

## CS LAB II : UNIX, WINDOWS NT AND DESIGN AND ANALYSIS OF ALGORITHMS

### UNIX

1. Writing payroll system using shell programming. ( Data input, validation, information queries, information output, database update, system interface )
2. Writing interactive commands to search through your directories removing junk files. Make sure that command runs after 10 p.m. to reduce system load.
3. Write a shell program to look through the arguments on command line and process them if they are files.
4. Write an infinite loop to prompt the terminal user for file names to be removed and remove them. Use TRAP to exit gracefully when finished.
5. Write a shell program to test for arguments of the form -c, -d, -e and so forth. Set arguments by the same names (c, d, e, ... ) to TRUE(1) or FALSE(0) depending on whether the argument exists on the command line.
6. Take any shell program and convert to shell function.
7. Create a reused catalogue using shell. As you create a catalogue, create a reusable shell component that comprise at least 10 % of the total catalogue.
8. Writing shell interface that would accept the file type, program name & source name and add the file to source code control system.
9. Creating a shell browser that will search by keyword.
10. Developing text editor using CURSES.H.
11. Practicals on utilities like AWK, NROFF, TROFF etc.

### WINDOWS - NT

Practicals on WINDOWS - NT specified by the course instructor.

### DESIGN AND ANALYSIS OF ALGORITHMS

1. Write recursive Pascal/C program for searching an element form a given array.
2. Write recursive Pascal/C program for finding maximum element form a given array.
3. Write recursive Pascal/C program for finding Binomial coefficient of two integers defined as  $B(n,m) = B(n-1,m-1)+B(n-1,m)$ ,  $B(n,n) = B(n,0)=1$ .
4. Write Pascal/C program for creating min. heap using INSERT.
5. Write Pascal/C program for creating max. heap using INSERT.
6. Write Pascal/C program for creating min. heap using ADJUST/HEAPIFY.
7. Write Pascal/C program for creating max. heap using ADJUST/HEAPIFY.
8. Write recursive, iterative Pascal/C modules in a program for finding greatest common divisor of two numbers.
9. Write divide & conquer based Pascal/C program for finding maximum, minimum element form a given array.
10. Write divide & conquer based Pascal/C program for searching an element form a given array using binary search for  $n = 1000, 2000, 3000$  find exact time of execution.
11. Write greedy based Pascal/C program for a solution of a knapsack instance.

12. Write divide & conquer based Pascal/C program for sorting elements of arbitrary array in ascending order using heap sort for  $n = 1000, 2000, 3000$  find exact time of execution.
13. Write divide & conquer based Pascal/C program for sorting elements of arbitrary array in descending order using heap sort for  $n = 1000, 2000, 5000$  find exact time of execution.
14. Write divide & conquer based Pascal/C program for sorting elements of arbitrary array in ascending order using merge sort for  $n = 1000, 2000, 3000$  find exact time of execution.
15. Write divide & conquer based Pascal/C program for sorting elements of arbitrary array in descending order using merge sort for  $n = 1000, 2000, 5000$  find exact time of execution.
16. Write divide & conquer based Pascal/C program for sorting elements of arbitrary array in ascending order using quick sort for  $n = 1000, 2000, 3000$  find exact time of execution.
17. Write divide & conquer based Pascal/C program for sorting elements of arbitrary array in descending order using quick sort for  $n = 1000, 2000, 5000$  find exact time of execution.
18. Write divide & conquer based Pascal/C program for multiplying two  $2 \times 2$  matrix using Strassen's matrix multiplication algorithm.
19. Write Pascal/C program for all pair shortest path problem using dynamic programming.
20. Write Pascal/C program for obtaining optimal code for a given expression using CODE1.
21. Write Pascal/C program for obtaining optimal code for a given expression using CODE2.
22. Write Pascal/C program for obtaining all solutions of 8 Queens problem using backtracking.
23. Write Pascal/C program for obtaining only inequivalent solutions of 4 Queens problem using backtracking.
24. Write Pascal/C program for obtaining all solutions of 10 Queens problem using backtracking, estimate time required by the algorithm.
25. Write Pascal/C program for UNION, FIND algorithm.
26. Write iterative Pascal/C program for searching an element from a given array, using rules of removal of recursion.
27. Write iterative Pascal/C program for finding maximum element from a given array, using rules of removal of recursion.
28. Write iterative Pascal/C program for finding Binomial coefficient of two integers defined as  $B(n,m) = B(n-1,m-1) + B(n-1,m)$ ,  $B(n,n) = B(n,0) = 1$ , using rules of removal of recursion.
29. Write Pascal/C program for Graph coloring algorithm.

## CS 201 COMPUTER GRAPHICS

**Graphics hardware :** Graphics input devices, graphic display devices and processors.

**Raster algorithm for 2-D output primitives :** Scan converting lines, circles and polygons. Polygon filling algorithms. Windowing and clipping ( point, line and polygon clipping ), attributes of the output primitives.

**General transformation :** Homogeneous coordinates, basic 2-D and 3-D transformation, composite transformations, modeling transformations, window to viewport transformations.

**3-D viewing :** General view specification, viewing transformation, mathematics of projections (parallel and perspective projection ), view-volume specification, clipping in 3-D.

**3-D object representation :** Polygonal models, curve and surface representation ( polynomial curves and surfaces, coons patches, hermit interpolation, Bazier curves and surfaces ). Fractals. Constructive solid geometric representation, boundary representation, sweep representation.

**Color in computer graphics :** chromatic and achromatic light, trichromatic theory, color models, color lookup tables, color quantization.

**Image synthesis :** illumination models, shading methods, ray tracing.

**Visual realism in computer generated images :** Visible surfaces and visible lines determination, shadows and transparency, aliasing, antialiasing and filtering, dithering and halftoning.

**User interface design :** logical input devices, interaction task, interaction modes, interaction techniques.

### Relevant Books :

1. Computer graphics by Baker.
2. Theory and problems of computer graphics by Plastock and Kalley.



## CS 202 PARALLEL COMPUTING

**Introduction :** Computational demand of modern science, advent of practical parallel processing, parallel processing terminologies : pipelining and data parallelism, control parallelism, scalability, The sieve of erathosthenes : control parallel approach, data parallel approach, data parallel approach with I/O.

**PRAM Algorithms :** PRAM model of parallel computation, PRAM algorithms : parallel reduction, prefix sums, list ranking, preorder tree traversal, merging two sorted list, graph coloring, reduce number of processors : problem defying fast solution on PRAM.

**Processor Arrays, multiprocessors and multicomputers :** processor organization : mesh networks, binary tree network, hypertree networks, pyramid networks, butterfly networks, hypercube networks, cube connected cycle networks, shuffle exchange network, de Bruijn network. Processor Arrays : Connection machine CM-200, multiprocessor : uniform memory access (UMA) multiprocessor, non uniform memory access (NUMA) multiprocessors, Multicomputers, Flynn's Taxonomy, Speedup, Scaled speedup, parallelizability

**Parallel programming languages :** Programming parallel processors, overview of FORTRAN 90, C\*, OCCAM, C-LINDA

**Solving linear systems :** Terminology, back substitution, odd-even reduction, Gaussian elimination, the Jacobi algorithm, Sparse's linear systems, Gauss-Siedal algorithm, Jacobi SOR method, MultiGrid method, Conjugate gradient methods.

**Parallel Sorting methods :** Quicksort based algorithm, Enumeration sort.

### Relevant Books :

1. Quinn M.J. : Parallel Computing : theory and practice. MGH
2. Lester B.P. : The art of parallel programming. PHI Eglewood Cliffs, N.J.
3. OCCAM reference manual, NMOS limited.

## CS 203 ARTIFICIAL INTELLIGENCE AND EXPERT SYSTEM DESIGN

**Problem Solving :** state space representation, heuristics, heuristic evaluation functions, problem reduction. Search methods : generate and test, hill climbing, means-ends analysis, depth-first, breadth-first, best-first, exploiting domain constraints, dependency-directed backtracking, minimax, alpha-beta pruning, iterative deepening, A\*, AO\*.

**General Issues in Knowledge Representation and Inference :** adequacy, richness, granularity, ease of representation and use, modeling uncertainty, the Frame Problem, declarative and procedural representations, representational equivalence.

**LOGIC :** propositional and predicate logic, representation - atoms, connectives, literals, CNF, DNF and clausal form; interpretation and model; satisfiability; resolution principle and unification.

**Rules :** working memory, rule base, conflict set, conflict resolution strategies (including specificity, recency, refractoriness), backward and forward chaining, meta-rules.

**Structured Representations :** semantic networks, frames, conceptual dependency, scripts, inheritance, default values, example systems, blackboards.

**Truth Maintenance Systems.**

**Uncertainty Reasoning :** probability theory, Bayesian networks, certainty, factor methods, basics of fuzzy logic, non-monotonic reasoning.

**Neural networks.**

### Application Topics

**Machine learning :** explanation based learning, rule induction, ID3 algorithm, version spaces

**Planning :** blocks world, STRIPS, hierarchical planning, non-linear planning, Sussman anomaly,

**Expert System :** rule based systems, MYCIN model of uncertainty, knowledge engineering, rule base verification, other approaches such as model based reasoning

Natural Language Understanding : Syntactic processing semantic analysis, pragmatics, case grammar, parsing, ATN, RTN

#### **Relevant Books**

1. Artificial Intelligence by E Rich and K Knight, McGraw Hill / Kogakusha Students Edition 1991.
  2. Artificial Intelligence (3rd edition) by PH Winston, Addison Wesley, 1992
  3. Introduction to Artificial Intelligence by E Charniak and D McDermott, Addison Wesley.
  4. Rule based expert System : A Practical Introduction by M Sasikumar, S Raman, S Muthu Raman, KSR Arjaneyulu and R Chandrasekar, Narosa Publishing HOUSE New Delhi and Addison-Wesley, Singapore 1993.
  5. LISP (3rd Edition) by PH Winston and BKP Horn. Addison Wesley, 1989.
- Introduction to Artificial Intelligence and Expert Systems by DW Patterson, Prentice-Hall 1990 (Eastern Economy Edition) -

### **CA 204 SOFTWARE ENGINEERING**

Software engg. Principles : how is software engg an 'engineering discipline?', information system characteristics, software development process models, life cycle concepts, software phases and deliverables, software development strategies.

Technical development : structured system analysis and design, requirements collections and specifications, data flow and logical data modeling, Cost Benefit analysis, feasibility study, Architectural and detailed design, process, data, network, control and user interface design, physical data design, dynamic modeling for real time systems

Software project management : principles of software project management, organizational and team structure, project planning, project initiation and termination, technical, quality, and management plans, project controls, cost estimation methods- function point and COCOMO, tools

Software quality management : quality control, quality assurance, quality standards, software metrics, verification and validation, testing, quality plans, tools

Configuration management

Software development methods and CASE : formal, semiformal and informal methods; data, function and event based modeling, some of the popular methodologies such as Yourdon's SAD, SSADM etc., CASE tools, CASE standards. Detail study of TURBO ANALYST

Implementation : in 3GL environments, in 4GL environments, in Client-server environments, coding styles, Documentation, Software maintenance.

#### **Relevant Books**

1. Software Engineering concepts by Fairley TMH
2. Software Engineering by Pressman Ed.III. TMH

### **CS 205 THEORETICAL COMPUTER SCIENCE**

Formal languages, phrase structured grammar, Type 0, type 1, type 2., type 3 languages and their examples, Chomesky hierarchy, Theory of context free grammar, Chomesky normal form and Greibach normal form, Formal models of computation, Finite automata, push down automata, turing machines, halting problem of T.M., correspondence between languages and automata.

Recursive and recursively enumerable sets, partial recursive function.

Introduction to Church-Godel hypothesis, Computability Turing computability.

#### **Relevant Books :**

1. Zohar Manna, Mathematical theory of computations, MGH (1974)
2. W. S. Brainerd and L.H. Landweber Theory of computations John Wiley, 1974
3. J. E. Hopcraft and J. D. Ullman, Formal Languages and their relation to automata, Addison Wesley, 1969

## CS LAB III : PARALLEL PROGRAMMING AND ARTIFICIAL INTELLIGENCE & EXPERT SYSTEM DESIGN

### PARALLEL PROGRAMMING

Multipascal implementation for solving

1. Parallel reduction algorithm
2. Prefix sum algorithm
3. List ranking algorithm
4. Merging two files
5. Graph coloring problem
6. Gaussian elimination
7. Jacobi algorithm
8. Sparse linear system
9. Gauss Seidel method
10. Multigrid method
11. Sorting algorithm

### ARTIFICIAL INTELLIGENCE AND EXPERT SYSTEM DESIGN

1. Implement Breadth-First Search algorithm in PROLOG/LISP language.
2. Implement Depth-First algorithm in PROLOG/LISP language.
3. Implement Heuristic Search algorithm in PROLOG/LISP language
4. Implement Generate-and-Test algorithm in PROLOG/LISP language
5. Implement Hill Climbing Search algorithm in PROLOG/LISP language
6. Implement Steepest-Ascent Hill Climbing Search algorithm PROLOG/LISP language
7. Implement Best -First Search algorithm in PROLOG/LISP language.
8. Implement A\* algorithm in PROLOG/LISP language
9. Implement AO\* algorithm in PROLOG/LISP language
10. Implement Constraint satisfaction algorithm in PROLOG/LISP language .
11. Implement Resolution algorithm in PROLOG/LISP language
12. Implement Breadth-First Search algorithm in PROLOG/LISP language
13. Case Study on Expert System.

## CS LAB IV : COMPUTER GRAPHICS AND CASE TOOLS

### COMPUTER GRAPHICS

Implementation of following algorithms in C/C++

1. Line drawing algorithm
2. Circle drawing algorithm
3. Ellipse drawing algorithm
4. Polygon filling algorithm
5. Windowing and clipping algorithm ( point, line & polygon clipping )
6. Composite 2-D transformation
7. 3-D curves and surface representation i.e. B-spline curves & surfaces, Polynomial curves & surfaces, Bezier curves and surfaces
8. 3-D geometric transformation ( Rotation, scaling and reflection ) and composite transformation
9. Determination of visible surfaces and lines
10. Combining all the modules to form a single graphics package.

## CASE TOOLS

Practicals on CASE TOOLS specified by the course instructor

## CS 301 COMPILER CONSTRUCTION -

Automata theory, Regular Expressions, NFA, DFA, Minimize DFA, Transition table different phases of compilation

Theory of parsing, top down and bottom up parsing methods, LR parsers

Intermediate code generation, different types of intermediate codes

Code Optimization techniques.

### Relevant Books :

1. Aho and Ulman, Principles of Compiler Construction.
2. BackHouse Principles of compiler construction.
3. Girs Compiler Construction

## CS 302 COMPUTER NETWORKS AND DISTRIBUTED COMPUTING

7 layer OSI architecture of ISO, concept of layer protocols and layer interfaces.

Physical layer concept of bandwidth, bit rate, error rate, transmission delay, modems, speed and compression standards.

LAN, WAN, MAN network topologies.

LAN standards- ETHERNET, token ring, token bus, MAC layer concept of LANs, FDDI, radio and satellite broadcast networks.

Data link and network layer Protocols and concepts, circuit switching, packet switching and message switching, datagram and virtual circuit, concepts of routing and congestion control.

Transport layer transport connections, understanding of different levels of transport level protocols and their applications.

Concepts of well known ports, initial connection problems, name/directory server.

Network application for file transfer and virtual terminals, knowledge of OSI application like VTAM, FTAM, X.400, X.500, client-server applications, network file server.

Knowledge of TCP-IP and X.25 levels 2 & 3, concept of router based networking, products like router, bridge, repeater gateway, modem.

Knowledge of UNIX networking concepts such as socket programming.

### Relevant Books

1. computer networks by A.S. Tanenbaum.
2. Martin, J : Computer Networks and Distributed Processing , PHI
3. Comer , Internetworking with TCP/IP : Principles , Protocol s, and architecture, PHI

## CS 303 COMPUTER BASED OPTIMIZATION TECHNIQUES

1. Linear programming : mathematical model, assumptions of linear programming, principles of simplex method, revised simplex method, applications, duality, dual simplex method, sensitivity analysis.

2. Special types of linear programming problems : transportation and assignment problems.

3. Decision theory and games.

4. Sequencing and scheduling PERT/CPM.

### Relevant Books :

1. Operations research by Taha H.A. McMillan
2. Introduction to operation research by Gillett. MGH
3. Operation research by Ravindran Wiley Eastern
4. Introduction to operation research by Hiller and Liberman. Holden Day Inc.

## CS 304 PRINCIPLES OF PROGRAMMING LANGUAGES

### SECTION A (Programming Paradigms)

**Object Oriented Programming** : Objects, Classes, messages, inheritance, binding, multiple inheritance, polymorphism.

**Functional Programming** : Functional approach, refinement, functions and classes, SUGAR.

**Declarative programming** : Declarative semantics, procedural semantics, soundness and SLD resolution, computation rule, refutation procedures, cuts, negation and completion.

**Procedural programming** : a review.

### SECTION B (Comparative Study of Programming Languages)

C++, LISP, PROLOG & C.

### SECTION C (Design and Implementation of a new programming language)

Data types, sequence control structure, procedures and functions, storage management etc.

#### Relevant Books :

1. Pratt : Programming languages : Design and implementation. PHI
2. Tucker : Programming Languages MGH
3. Horowitz, Sahani : Programming Languages . Galgotia Publi.

## CS 305 CURRENT COMPUTING TRENDS

The current softwares are to be taught .

### CS LAB V : VISUAL BASIC AND WINDOWS PROGRAMMING AND OPTIMIZATION TECHNIQUES

#### VISUAL BASIC

1. Understanding the concepts of visual programming.
2. Understanding the concepts of **object, forms & control classes**.
3. Designing an application illustrating the use of following.

#### All control classes ( Command button, option button etc. )

4. Developing an application to demonstrate following
  - I] Manipulating objects on different forms.
  - II] Event model of Visual BASIC.
5. Developing a graphical editor.
6. Using menu designer.
7. Understanding use of Visual BASIC as front-end tool. ( Concept of ODBC).
8. Manipulation of tabledef & querydef objects.

#### WINDOWS PROGRAMMING

1. Write windows program to display windows messages in a scrollable window.
2. Write a windows program which just passes all the messages to `defwindowproc()` except one.
3. Explore the concept of device contexts by writing simple window program.
4. Write simple window program for displaying text on window at center ( use different fonts for text).
5. Write windows program obtains information about character and display all this in its window.
6. Write windows program which display scrollbar messages in a scrollable window.
7. Study different windows keystroke messages and write windows program to display all keyboard character messages in a scrollable window.
8. Study different mouse messages and write window program to display it in a scrollable window.
9. Study different window activation messages and write window program to test their functions.
10. Study concept of mouse capturing and write windows program to demonstrate mouse capturing.
11. Write windows program utility which allows you to view the hexadecimal display of any given file in a movable, resizable window also add common dialog box for opening file.

12. Study different predefined child window control classes and design a simple user interface using these control. (Static, Button, Edit, List box, Scrollbar).
13. Write windows program to create windows menu (Menus, Popupmenus) using the API.
14. To study use of menu template to create menus.
15. Write windows program for demonstrating printing under windows.
16. Write windows program which demonstrate the various method of font selection and their effects.
17. Write windows program for displaying windows device driver capabilities(Use Scrollbars.)
18. Write windows program which demonstrates use of following function.
  - I) Setpixel()
  - II) Lineto(HDC, int, int)
  - III) Moveto(HDC, int, int)
  - IV) Settextalign
  - V) Creatpen(int, int, COLORREF)
  - VI) Rectangle
  - VII) RoundRect
  - VIII) FillRect
  - IX) InsertRect
  - X) FrameRect
  - XI) Ellipse
  - XII) Arc
  - XIII) Chord
  - XIV) Create Solid Brush

#### OPTIMIZATION TECHNIQUES

Develop programs in C/C++ for the following

1. Graphical method to solve L.P P
2. Simplex algorithms
3. Revised simplex method
4. Dual simplex
5. Transportation problem
6. Assignment problem
7. Solving NxN game

### CS LAB VI : COMPILER CONSTRUCTION, VISUAL C++ AND NETWORK PROGRAMMING

#### COMPILER CONSTRUCTION

1. Write a program to test whether or not a given string can be accepted by a given DFA
2. Write a program to convert NFA into DFA & minimize it.
3. Write a program for simple shift reduce parser using stack.
4. Write a program for recursive descent parser
5. Write a program for LL parser.
6. Write a program for SLR parser.
7. Write a program for Operator Precedence Grammar.

#### VISUAL C++

1. Understanding the use of AppStudio as a prototyper
2. Understanding the use of common user interface element ( such as common dialogues, file open dialogue, box, choosing document type (SDI/MDI) )
3. Designing new classes using MFC
4. Understanding how document and view architecture works as well as their printing.
5. Drawing in the nonclient area of a window and understanding how to hook up messages to classes in MFC
6. Creating popup menu
7. Understanding concepts of window hooks
8. Understanding concepts of Dynamic Linking (DLL) and Static Linking

9. Developing application which allows you to do

- Hook sounds to various windows system events, text buttons, and window action
- Animate your icons
- Animate your cursor
- Create and edit your own icons, cursors and small bitmaps
- Provide a popup program launcher

( Create DLL for this application )

10. Understanding concepts of OLE and use OLE features for developing application

11. Understanding use of MFC for creating ODBC data sources.

12. Understanding the use of database features of MFC classes.

**NETWORK PROGRAMMING**

1. Writing routine for printing across the LAN.
2. Writing transaction tracking system.
3. Writing routine in c/c++ for file sharing.
4. Writing routine in c/c++ for record locking.
5. Writing driver for printing to a shared printer using underlying operating system services. ( main concerns are the mode, fonts or typeface pitch and other printer characteristics used for your report, the possible insertion of stray page breaks into the report by the network operating system.)
6. Writing routine for peer - level communication on a LAN (message passing ) using NETBOIS , SPX/IPX, TCP/IP.
7. Writing Remote Program Execution Utility.
8. Writing Electronic Mail utility in c/c++.

**SEMESTER FOUR :**

**Full time Six month Industrial Training.**