M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I

PGCS-101: Algorithm Design Patterns

Course Outcomes:

At the end of the course, a student shall be able to:

- understand different algorithm design techniques and their features
- create efficient algorithms for real world problems
- implement techniques for synthesis & analysis of algorithms

Total Credits: 04 **Total lectures: 60**

Course content

1 .Introduction: (4L)

Algorithm Concept, Writing structured programs, Analyzing algorithms

2. Divide & conquer: (10L)

The General method, Binary search, Finding the maximum &minimum selection, merge sort

3. The greedy method: (12L)

The general method, Optimal storage on tapes, Knapsack problem, Job Sequencing with deadlines, Optimal merge patterns, Minimum spanning tree

4. Dynamic programming: (12L)

The general method, Multistage Graphs, All pairs shortest path, Optimal binary search tree, Flow shop scheduling

5. Basic search & traversal techniques: (12L)

The techniques, code optimization, AND/OR graphs Game tree, Backtracking. The general method, The 8-queens problems, Sun of subsets, Graph coloring, Hamilton cycles

6. Branch-and-Bound: (6L)

Strategy ,The method 0/1 knapsack problem, Traveling salesperson, LIFOBB, FIFOBB

7. NP-HARD&NP- COMPLETE problem:

(4L)

Basic concepts, cook's theorem, NP-HARD graph problem, NP-HARD scheduling problem

References Books:-

1) Fundamentals of computer Algorithms – Coreman
2) Algorithm Design – Kleinberg and Tardos

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I
PGCS-102: Paradigms of Programming Languages

Course Outcomes:

At the end of the course, a student shall be able to:

- think about programming languages analytically andseparate syntax from semantics.
- develop a greater understanding of the issues involved in programming language design and implementation.
- familiar with design issues of object – oriented and functional languages.
- analyze semantic issues associated with function implementations, including variable binding, scoping rules, parameter passing.

Total Credits: 04 **Total lectures: 60**

Course content

1. **Introduction to Programming languages: (6L)**
   Need for studying programming languages, A Short of programming languages, Features of a good language, Effects of environments on languages
2. **Principles of languages: (10L)**
   Design Structure and operation of a computer Virtual computers and languages
   Implementations syntax, type and semantics, Context free grammars, Grammar For Expression, Lexical analysis, parsing
3. **Paradigms of Programming language**: **(10L)**
   Imperative paradigms (simple Procedural languages) Fortran, C. Block structured paradigm Pascal. The object based paradigm-Ada, C++, small talk, Logic programming paradigm-Prolog, The database language paradigm, declarative paradigm, Event driven programming, Fourth generation languages
4. **Building blocks of a language : (8L)**
   Data Object, variable constants, Data types- A brief classifications, Derived, Abstract, User- Defined Type conversion
5. **Procedures: (8L)**
   Designing of procedure, Simple call return sub-program, Sub-programs, subroutines, Recursive subprogram, Referencing environment-local And global Different parameter passing call-By-value, call-by-reference
6. **Functional programming: (10L)**
   Characteristics of functional Programming, Elements of Functional Programming language, Functions, Functional forms, Functional declarations , Expression Evaluation Logic programming, Proofs, facts, queries, logical Variable, substitutions Instances.
7. **Object oriented programming: (10L)**
   Introduction Basic concepts-object, class, Characteristics of OOP, Abstraction, Encapsulation and information, Hiding, Inheritance, Multiple inheritance, Polymorphism, OOP concepts in C++
8. **Comparative study of programming languages: (4L)**
   Control structures Flow control, rules, Basic control structures

Reference Books :

1. Programming languages design $ implementation by terrace W.Pratt, Marvin V Zelkowit
2. Programming paradigms Methology by G.P.Potdar

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I

PGCS-103: Advanced Database Concepts

Course Outcomes:

At the end of the course, a student shall be able to:

- translate complex conceptual data models into logical and physical database designs.
- design high-quality relational databases and database applications.
- understand location, replication and fragmentation independence
- implement advance database concepts and techniques

Total Credits: 04 **Total lectures: 60**

Course content

Object Oriented Databases:- **(12L)**

Overview of Object Oriented Concept ,Object Identity ,Object Structure , Object Definition Language, Types of Constructors , object database conceptual design, OODBMS advantages, Object query language examples of OODBMS

Distributed Databases : **(12L)**

Introduction to Distributed data processing, Homogeneous and heterogeneous systems ,Distributed DBMS Architecture, fragmentation , Distributed database design , Overview of Query processing, Query decomposition and data localization

Distributed DBMS reliability: **(12L)**

Reliability concepts & measures, Failures & fault tolerance in distributed systems, Local reliability protocols, Distributed reliability protocols, Network partitioning

Parallel Databases: **(12L)**

Parallel database concepts, Parallel database system architecture, Query parallelism, parallel Data processing and parallel query optimization

Emerging Database Technologies & Applications: **(12L)**

Multimedia Databases, Mobile Databases, Geographical Information Systems, Spatio – temporal patterns, Intervals and scalar operators in temporal databases, web databases,
Deductive Databases

Reference Books:-

1. Fundamentals of Database Management Systems By Navathe & Elmasri (3rd Edition, Pearson Education)
2. Principles of Distributed Database Systems by Patrick Valduriez (3rd Edition' Pearson Education)
3. Introduction to Database Systems by C G Date (7th Edition Pearson Education)

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I

PGCS – MI: Minor Project-I

Learning outcomes:

At the end of the course, a student shall be able to:

- apply knowledge and techniques learnt in theoretical classes for developing the s/w for real problems.
- Get an insight into the working of the real organizations/companies.
- gain deeper understanding in specific functional areas.
- exploring career opportunities in their areas of interest.

Total Credits: 04

Course content

The course Minor Project is one that involves requirement analysis, feasibility analysis, Database design, coding, testing, implementation and maintenance.

Student will select individually Commercial or Technical project based on Technologies learnt in Semester I. Each student will have to prepare proper documentation consisting of SRS, Modeling Techniques, Development Strategies and Implementation and Testing Strategies. Student may use any Design Methodologies such as SSAD, OOAD and UML etc.

This is a documentation project only. The project work will be presented by student using Power Point Presentation. The Institute may appoint external expert from industry or academics if it feels so. The students will be assessed internally by such panel for this project.

- The Project can be platform, Language and technology independent.
- Project will be evaluated by project guide.
- Assessment will be done weekly in the respective batch.
- Evaluation will be on the basis of weekly progress of project work, progress report, oral, results and documentation and demonstration.
- You should fill your status of the project work on the progress report and get the Signature of project guide regularly.
- Progress report should sharply focus how much time you have spent on specific task. (The format of progress report is given as follow.)
- You should keep all signed progress reports.
- Project will not be accepted if progress report is not submitted and all responsibility remains with student.
- Students should prepare design document using SE/UML techniques depends on your project.

About project Report:

The report should be typed on A4 size, executive bond paper for the final submission. The report should be in the good quality Rexene bound. We suggest, using one-and-half spaced printing, Times New Roman 12 font sizes for the normal text, 14-16 font sizes for headings & page titles. Number of copies: For one project you should prepare 2 copies of the project report. One for yourself, one for college (College copy can be in CD).

Evaluation for internal 40 marks

Description	Marks
UML /ERD/DFD diagrams	10
Technology and design base first demo	10
Project technology based two assignments	10
Second Demo	10

Evaluation for external 60 marks

Description	Marks
Demo	10
Report	10
Presentation	20
Viva	20

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I

Core Elective-I PGCS-104: Parallel Processing

Course Outcomes:

At the end of this course, a student shall be able to :

- understand the fundamental aspects of parallel processing,
- familiar with taxonomies of parallel system,
- familiar with performance measures for parallel system,
- understand the theoretical limitations of parallel computing,
- write efficient parallel application programs.

Total Credits: 04 **Total lectures: 60**

Course content

1. Overview:
Motivation of parallelism, Scope of parallel Computing, Current trends in parallel processing **(4L)**

2. Concept Parallel Machine Model: **(6L)**
Parallel Programming Model, Parallel algorithm examples

3. Designing Parallel Algorithms: **(4L)**
Methodical Design, Partitioning, Communication, Agglomeration, Mapping, Case Studies

4. Quantitative basis for design: **(4L)**
Defining & Modeling performance, Developing Models, Scalability analysis, Communication cost model, Case Studies

5. Modular Designs: Modularity & parallel computing Case studies **(4L)**

6 .Tools: **(12L)**
Compositional C++ ,C++ review ,C C++: Introduction C++: Communication Case studies, Fortran M, Introduction, Communication, Argument passing, mapping, modularity, performance issues ,High performance Fortran, Data parallelism, data distribution & concurrency, Modularity, other HPF features & performance issues

7. Message passing interface: **(8L)**
MPI models & MPI basics, Communication, modularity & other features

8. Performance tools : Performance analysis, Tools related to performance analysis **(6L)**

10.Hypercube algorithm: Vector reduction & matrix transposition, Merge sort **(4L)**

11.Summary & revision: Parallel & distributed computing, Programming & analytical tools **(4L)**

Textbooks:

1. Designing &building parallel programs (2004) by I. Foster. Addison Wesley, ISBN 0-201-57594-9
2. Introduction to Parallel Computing, 2nd ed.(2003) by Ananth Grama, Anshul Gupta, George Karypis & Vipin Kumar. Benjamin/Cummings, ISSBN 0-8053-3170-0.

Reference Books:

3. Introduction to Parallel Processing by Prakash P. Ravi, Sasikumar M. & Shikhare Dinesh. Prentice Hall of India
4. Practical parallel programming by Wilson Gregory V. Prentice Hall of India
5. Parallel computer architecture (1999) by D.Culler & J.P.Singh. Morgan Kaufmann
6. Distributed & parallel computing (1997) by h. El-Rewini & t.g.Lewis
7. parallel programming with MPI(1996) by peter Pacheco Barnes & noble

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I
Core Elective-I PGCS-105: Theory of Automata

Course Outcomes:

At the end of this course, a student shall be able to :

- understand the core concepts in automata theory and formal languages.
- design grammars and automata (recognizers) for different language classes.
- identify formal language classes and prove language membership properties.
- to prove and disprove theorems establishing key properties of formal languages and automata.
- understand core concepts relating to the theory of computation and computational models

Total Credits: 04 **Total lectures: 60**

Course content

1. Introduction: Review of Mathematical, Preliminaries, Relations, Functions, Set theory, and Predicate and Prepositional, Calculus, principle of Mathematical induction /strong, Mathematical induction **(4L)**

2. Finite state machine: Definition, finite control, transition graphs, adjacency matrix, finite automata, determinist finite automata, language acceptance by FA, Moore and Mealy Machines, Finite state machine with output, FA MINIMIZATION AND RELATED THEOREM. **(10L)**

3. Regular Expression: Recursive regular expression, regular set, NFA with E moves, NFA without E moves, Inter conversion between NFA and DFA, Regular expression and FA, pumping LEMMA **(8L)**

4. Grammar: Context free grammar and it's properties, derivation tree, simplifying CFG, unambiguous CFG, CNF, GNF of CFG, Chomsky hierarchy, Chomsky normal form, derivation graphs type 0 and type 1 grammar. Concept of linear bounded automata, context sensitive grammars and their equivalence **(8L)**

5. Push down automata: Push down automata 2 way PDA, relation of PDA with CFG, deterministic and nondeterministic PDA and related theorems **(6L)**

6. Turing Machine: Definition model, comparison of Turing machine, example of TM, universal TM, tm limitation, church's Turing hypothesis ,multi stack TM ,halting problem, unrestricted grammars and their equivalence with TM, determinism and non-determinism of TM, TM as acceptor /generator ,algorithms and related theorems multi-tape, multi-head, multi-stack TM **(12L)**

7. Complexity: Introduction recursively numerable, sets, recursive set, partial recursive sets, Russell paradox undesirability and some non-computable problems **(8L)**

8. Applications: Application of RE and PA – lexical analyzer, text editor and searching using RE, application of PDA expression conversion, application on CFG, syntax analysis **(8L)**

Reference Books:

1. Daniel LA Cohen, "Introduction to computer theory". Wiley Publication.
2. John C. Martin, "Introduction to language and theory of computation", Mc Grawhill.
3. HopCroft Ullman,"Introduction to automata Theory, Language and Computations", Narosa.
4. Hapcraft Ulman "Introduction to Automata Theory".
5. Harry R. Lewis,"Elements of Thoery Of Computation"
6. E.V. Krishnamurthy "Theory of computer science" EWP publications.
7. LIU.C.L., Elements of Discrete Mathmatics, Mc Grawhill.
8. Aho,Ulman,Sethi "Principles of compiler construction"

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I

Core Elective-I PGCS-106: **Digital Image Processing**

Course Outcomes:

At the end of the course, a student shall be able to:

- analyze general terminology of digital image processing.
- examine various types of images, intensity transformations and spatial filtering.
- develop Fourier transform for image processing in frequency domain.
- evaluate the methodologies for image segmentation, restoration etc.
- implement image process and analysis algorithms.
- learn different feature extraction techniques for image analysis and recognition

Total Credits: 04 **Total lectures: 60**

Course content

1. **What is Digital Image Processing: (4L)**

Low level image processing, High level image processing, The origins of Digital Image Processing, Examples of fields that use Digital Image Processing, X-Ray Imaging, Imaging in the Ultraviolet Band, Visible and Infrared Band, Microwave Band, Radio Band, Fundamental steps in Digital Image Processing, Components/Elements of Digital Image Processing

2. **Elements of Visual perception: (6L)**

Structure of Human Eye, Image formation in the Eye, Brightness, Adaptation and Discrimination, Light and electromagnetic spectrum, Image sensing and acquisition, Image acquisition using a single sensor, Image acquisition using sensor strips ,Image acquisition using sensor array, A single image formation model, Image sampling and quantization, Basic concepts of sampling and quantization, Spatial and gray level resolution, Aliasing and moiré patterns, Zooming and shrinking of digital image, Some basic ,relationship between pixels, Neighbors of a pixel, Adjacency, Connectivity, Regions, Boundaries, Distance measures, Linear and non linear operations

3. **Image Enhancement in the Spatial domain**: **(10L)**

Introduction, Some basic grey level transformation Image negatives ,Log transformation, Power law(Gamma) transformations ,Piece wise linear transformation functions, Histogram processing, Histogram equalization ,Histogram matching(specification),Local enhancement, Image enhancement using arithmetic and logical operation, Image substation, Image averaging, Basics of spatial filtering, Smoothing spatial filters, Order statistic (non linear) filters, Sharpening spatial filters, Use of second derivative for enhancement-the Laplacian, Use of first derivatives for enhancement – the gradient, Combining spatial enhancement methods

4. **Image Enhancement in the Frequency domain: (10L)**

Introduction to the Fourier Transform and the Frequency Domain, One Dimensional Fourier Transform,2-D Discrete Fourier Transform (DFT)and its Inverse, Filtering in the Frequency Domain, Correspondence between Filtering in the Spatial and frequency domain, Smoothing Frequency –Domain Filters

5. **Image Restoration: (12L)**

Introduction, Noise Models, Gaussian Noise, Rayleigh Noise, Erlang Noise ,Exponential noise, Uniform noise, Impulse Noise, Periodic noise, Restoration in the presence of noise only spatial filtering, Periodic noise reduction by frequency domain filtering, Band reject alters, Bandpass filters, Notch filters, Estimating the Degradation Function, Estimation of

Degradation Function by Image Observation, Estimation of Degradation Function by Experimentation, Estimation of Image degradation by Modeling Geometric mean Filter, Inverse Filtering, Minimum Mean square error(Wiener)filtering ,Geometric Transformation

6. Morphological Image Processing: (8L)

Some basic concepts from set theory, Reflection and Translation, Logic operation involving Binary Images, Erosion and Dilation, Erosion, Dilation Duality, Opening and Closing, The Hit –or-Miss Transformation

7. Image Segmentation Fundamentals: (4L)

Detection of Discontinuities, Thresholding , Region –based Segmentation

8. Representation and Description: (6L)

Representation, Simple Boundary Descriptors, Simple Regional Descriptors, Use of principal components for description, Relational Descriptors

Reference books

1. Fundamental of Digital Image Processing by Anil k. Jain Prentice -Hall
2. Digital Image Processing by R.C. Gonzalez and Woods
3. Digital Image Processing using Matlab by R.C. Gonzalez and Woods,S .L Eddnins
4. Digital Image Processing PIKS Scientific Inside by William .K.Pratt ,WILEY India
5. Digital Image Processing and Computer Vision by Sonka ,Hlavac,Boyle, CENGAGE Learning

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I

PGCS – 107: Lab Course - I

Course Outcomes:

At the end of this course, a student shall be able to:

- understand and explain different data structures and their features
- explore various applications of arrays, stack and queues and linked list
- implement Btrees, AVL Trees, sorting and searching algorithms
- design and execute programs using linked representation of stack and queue

Total Credits: 02

Course content

List of' programming assignments to be executed in C / C++:

1. Represent sparse matrix using array and perform matrix addition or simple and fast transpose.
2. Represent polynomial as a circular linked list and write a menu driven program to perform addition, multiplication and evaluation.
3. Write a menu driven program to perform following operations on doubly linked list: Create, Insert, Delete and Display.
4. Create two singly or doubly linked lists, sort them after creation using pointer manipulation. Merge these two lists into one list without creating, a new node. Merged list should be a sorted one.
5. Write a program to create a generalized linked list and perform following operations copy, equivalence and depth.
6. Implement Stack as an abstract data type-using array or linked list. Use this ADT for expression conversion and evaluation.
7. Represent circular Queue using, array and write a pro-ram to perform following operations Insert, Delete, Finding front and rear element.
8. Creation of binary tree and perform recursive and non-recursive traversals.
9. Creation of binary inorder threaded tree and perform all three traversals.
10. Represent a given graph using adjacency list and perform DFS and BFS.
11. Represent a (given graph using adjacency list or array and find the shortest path using Dijkstra algorithm.
12. Represent a given graph using adjacency list or array and (generate a minimum spanning tree using Kruskal's and Prime's algorithms.
13. Implement binary search tree as an abstract data type.
14. Create a binary search tree and find height of a tree and print the leaf nodes.
15. Create a binary search tree, find its mirror image, Print original and mirror image using level wise printing.
16. Create a hash table and handle the collisions using linear probing with perform or without replacement.

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I

PGCS – 108: Lab Course - II

Course Outcomes:

At the end of the course, a student shall be able to:

- understand and implement C ,C++ programs logic.
- Implement the procedure and object oriented programming.
- develop applications using java and C# programming.
- apply file handling application through C, java and C# programming Languages.

Total Credits: 02

Course content

Any 16 assignments on the following topics

1. Basic programming structure using C,C++,JAVA,C#
2. Control structures: If statements, Loop structure
3. Programs based on Arrays: 1D,2D
4. String Manipulation
5. Functions: built in functions and user defined functions
6. Inheritance: single inheritance, multiple, multilevel and hybrid
7. Polymorphism using C++ and JAVA
8. Constructor
9. File handling

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I

PGCS – 109: Lab Course - III

Course outcomes:

At the end of this course, a student shall be able to:

- understand and enhance the fundamental concepts of SQL.
- explore the knowledge about SQL environment by performing DDL and DML operations.
- analyze test results of given experiments work with operations of SQL

Total Credits: 02

Course content

Note: My Access / MySQL may be used.

- **DDL Commands**
    - Create table, alter table, drop table
- **DML Commands**
    - Select , update, delete, insert statements
    - Condition specification using Boolean and comparison operators (and, or,
- not,=,<>,>,<,>=,<=)
    - Arithmetic operators and aggregate functions(Count, sum, avg, Min, Max)
    - Multiple table queries (join on different and same tables)
    - Nested select statements
    - Set manipulation using (any, in, contains, all, not in, not contains, exists, not
- exists, union, intersect, minus, etc.)
    - Categorization using group by……..having
    - Arranging using order
- Procedures
- PL/SQL block
- Triggers

Overall, 16 assignments should be conducted.

*** * * * * ***