NORTH MAHARASHTRA UNIVERSITY, JALGAON

School of Computer Sciences

MASTER OF SCIENCE in Computer Science
[M. Sc.(Computer Science)]

Syllabus
[under Academic Flexibility]

Faculty of Science and Technology

With effect from July- 2017-18

School of Computer Sciences
School of Computer Sciences
NORTH MAHARASHTRA UNIVERSITY, JALGAON – 425 001

COURSE STRUCTURE WITH CREDIT
[under Academic Flexibility]

M.Sc. (Computer Science) w.e.f. June 2017

<table>
<thead>
<tr>
<th>Semester</th>
<th>Course</th>
<th>Marks</th>
<th>Hour/week</th>
<th>Credit</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semester-I</td>
<td>CS-101</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>CS-102</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS-103</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS-104</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS-105</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS LAB-I</td>
<td>100</td>
<td>06</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS LAB-II</td>
<td>100</td>
<td>06</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>Semester-II</td>
<td>CS-201</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>CS-202</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS-203</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS-204</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS-205</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS LAB-III</td>
<td>100</td>
<td>06</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS LAB-IV</td>
<td>100</td>
<td>06</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Audit Course-I</td>
<td>50</td>
<td>02</td>
<td>02</td>
<td></td>
</tr>
<tr>
<td>Semester-III</td>
<td>CS-301</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>CS-302</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS-303</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS-304</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS-305</td>
<td>100</td>
<td>04</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS LAB-V</td>
<td>100</td>
<td>06</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CS LAB-VI</td>
<td>100</td>
<td>06</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Audit Course-II</td>
<td>50</td>
<td>02</td>
<td>02</td>
<td></td>
</tr>
<tr>
<td>Semester-IV</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>12</td>
</tr>
</tbody>
</table>

Industrial Training/Project
Audit Course I and II: To be completed using IIT Bombay’s SPOKEN TUTORIAL online Courseware. It is mandatory for the students to complete the credits of this course along with passing the online examination conducted by Spoken Tutorial IIT Bombay.

<table>
<thead>
<tr>
<th>Degree Name</th>
<th>Master of Science in Computer Science [M. Sc. (Computer Science)] [Under Academic Flexibility]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Faculty</td>
<td>Science and Technology</td>
</tr>
<tr>
<td>Duration</td>
<td>02 years, Full Time course</td>
</tr>
<tr>
<td>Medium of Instruction</td>
<td>English</td>
</tr>
<tr>
<td>Pattern</td>
<td>Semester Pattern (04 semesters)</td>
</tr>
<tr>
<td>Examination Pattern</td>
<td>60% (External Assessment) + 40%(Internal Assessment)</td>
</tr>
<tr>
<td>Passing Standard</td>
<td>Separate Passing for internal as well as external assessment.</td>
</tr>
<tr>
<td>Evaluation Mode</td>
<td>CGPA</td>
</tr>
<tr>
<td>Lecture</td>
<td>Clock hour (60 minutes)</td>
</tr>
</tbody>
</table>
# NORTH MAHARASHTRA UNIVERSITY, JALGAON
## School of Computer Sciences
### SYLLABUS for M. Sc. (Computer Science)
(With effect from June-2017)

#### Semester-I

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS-101</td>
<td>Data Structures and Algorithms</td>
</tr>
<tr>
<td>CS-102</td>
<td>Database Management System (DBMS)</td>
</tr>
<tr>
<td>CS-103</td>
<td>Automata Theory and Computability</td>
</tr>
<tr>
<td>CS-104</td>
<td>Operating Systems</td>
</tr>
<tr>
<td>CS-105</td>
<td>Object Oriented Programming using JAVA</td>
</tr>
<tr>
<td>CS LAB-I</td>
<td>LAB on Data Structures and Algorithms and JAVA programming</td>
</tr>
<tr>
<td>CS LAB-II</td>
<td>LAB on DBMS</td>
</tr>
</tbody>
</table>

#### Semester-II

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS-201</td>
<td>Compiler Construction</td>
</tr>
<tr>
<td>CS-202</td>
<td>Mathematical Foundations of Computer Science</td>
</tr>
<tr>
<td>CS-203</td>
<td>Artificial Intelligence</td>
</tr>
<tr>
<td>CS-204</td>
<td>Design and Analysis of Algorithms</td>
</tr>
<tr>
<td>CS-205</td>
<td>Python Programming</td>
</tr>
<tr>
<td>CS-206</td>
<td>Audit Course-I</td>
</tr>
<tr>
<td>CS LAB-III</td>
<td>LAB on Design and Analysis of Algorithms(DAA)</td>
</tr>
<tr>
<td>CS LAB-IV</td>
<td>LAB on Python Programming</td>
</tr>
</tbody>
</table>

#### Semester-III

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS-301</td>
<td>Machine Learning</td>
</tr>
<tr>
<td>CS-302</td>
<td>Natural Language Processing</td>
</tr>
<tr>
<td>CS-303</td>
<td>Computer Graphics and Digital Image Processing</td>
</tr>
<tr>
<td>CS-304</td>
<td>Software Engineering</td>
</tr>
<tr>
<td>CS-305</td>
<td>Web Application Development Technology</td>
</tr>
<tr>
<td>CS-306</td>
<td>Audit Course-II</td>
</tr>
<tr>
<td>CS LAB-V</td>
<td>LAB on Web Application Development Technology</td>
</tr>
<tr>
<td>CS LAB-VI</td>
<td>LAB on Computer Graphics and Digital Image Processing</td>
</tr>
</tbody>
</table>

**Note: Audit Course I and II:** To be completed using IIT Bombay’s SPOKEN TUTORIAL online Courseware. It is mandatory for student to complete the credits of this course along with passing the online examination conducted by Spoken Tutorial IIT Bombay.
Semester-IV

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS-401</td>
<td>Full Time Industrial Training</td>
</tr>
</tbody>
</table>
Course Code: CS-101 Data Structures and Algorithms

Clock Hours: 60
Total Marks: 100

Semester-I

Unit-I
Introduction to Data Structures and Algorithms: Algorithmic Notation: Format Conventions, Statement and Control Structures. Time and Space Analysis: Data types and Abstract data types, Types of Data structures; Primitive, Non primitive, Linear and Nonlinear Data structures

Max Marks: 06

Unit-II
Array: Storage representation, operations and applications (Polynomial addition and subtraction) Stack: operations and applications (infix, postfix and prefix expression handling), Queue: operations and applications, Circular Queues: operations and applications, Concept of Double ended Queue and Priority Queue, Linked representation of stack and queue.

Max Marks: 15

Unit-III
Linked Lists: Operations and Applications of Linear linked list (Polynomial addition and subtraction), Circular linked list and Doubly linked list.

Max Marks: 12

Unit-IV
Trees: Binary Trees, Binary Tree: Representations, Operations (insert/delete), Traversal (inorder, preorder, postorder, level order), Threaded Binary Tree, Search Trees: AVL Tree, single and double rotations, B-Trees: insertion and deletion, Introduction to B+ and B* Trees

Max Marks: 21

Unit-V
Graphs and Their Applications: Representation (Matrix/Adjacency) and Traversal (Depth First Search/Breadth First Search), Spanning Trees, Minimal Spanning Tree (Prim’s and Kruskals’s algorithm), Shortest Paths and All Pair Shortest Path, Dijkstra’s, Floyd-Warshall Algorithms.

Max Marks: 18

Unit-VI

Max Marks: 18
References:

Course Code: CS-102 Database Management System (DBMS) Clock Hours: 60
Total Marks: 100

Unit-I [05] Max Marks:08
Introduction: Database system application and purpose, Characteristics of DBMS, Database Users, 1-tier, 2-tier and 3-tier architecture of DBMS along with its advantages, Levels of Database Architecture, Data Models, Data-schemas and instances, Data Independence, Role and responsibilities of DBA.

Unit-II [10] Max Marks:10

Unit-III [12] Max Marks:20
Domain Relational Calculus.

**Unit-IV**


**Unit-V**

**Concurrency Control and Recovery System:** Lock based Protocol, Timestamp based Protocol, Validation based Protocol, Deadlock Handling, Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithms, Buffer Management, Early lock release and logical undo operations, Remote Backup Systems. Case study: ARIES

**Unit-VI**


**References:**


Course Code: CS-103  Automata Theory and Computability  

Clock Hours: 60  
Total Marks: 100

Unit-I  
[10] Max Marks: 20
**Grammars:** Production systems, Chomskian Hierarchy, Right linear grammar and Finite state automata, Context free grammars, Normal forms, uvwxy theorem, Parikh mapping, Self-embedding property, Subfamilies of CFL, Derivation trees and ambiguity

Unit-II  
[10] Max Marks: 20
**Finite State Automata:** Nondeterministic and deterministic FSA, NFSA with ε-moves, Regular Expressions, Equivalence of regular expression and FSA, Pumping lemma, closure properties and decidability, Myhill - Nerode theorem and minimization, Finite automata with output

Unit-III  
[08] Max Marks: 15
**Pushdown Automata:** Acceptance by empty store and final state, Equivalence between pushdown automata and context-free grammars, Closure properties of CFL, Deterministic pushdown automata

Unit-IV  
[12] Max Marks: 20
**Turing Machines:** Techniques for Turing machine construction, Generalized and restricted versions equivalent to the basic model, Godel numbering, Universal Turing Machine, Recursively enumerable sets and recursive sets, Computable functions, time space complexity measures, context sensitive languages and linear bound automata

Unit-V  
[08] Max Marks: 10
**Decidability:** Post's correspondence problem, Rice's theorem, decidability of membership, emptiness and equivalence problems of languages

Unit-VI  
[10] Max Marks: 15
**Complexity Measures:** Time and tape complexity measures of Turing machines, Random access machines, the classes P and NP, NP-Completeness, satisfiability and Cook's theorem, Polynomial reduction and some NP-complete problems, Regulated rewriting L systems, Grammar systems

*References:*
Unit-I [04] Max Marks:08
Introduction: review of computer organization, introduction to popular operating systems like UNIX, Windows, etc., OS structure, system calls, functions of OS, evolution of OSs.

Unit-II [03] Max Marks:06
Computer organization interface: using interrupt handler to pass control between a running program and OS.

Unit-III [08] Max Marks:12
Concept of a process: states, operations with examples from UNIX (fork, exec), Process scheduling, inter-process communication (shared memory and message passing), UNIX signals.

Unit-IV [04] Max Marks:06
Threads: multithreaded model, scheduler activations, examples of threaded programs.

Unit-V [06] Max Marks:10
Scheduling: multi-programming and time sharing, scheduling algorithms, multiprocessor scheduling, thread scheduling (examples using POSIX threads).

Unit-VI [08] Max Marks:12
Process synchronization: critical sections, classical two process and n-process solutions, hardware primitives for synchronization, semaphores, monitors, classical problems in synchronization (producer-consumer, readers-writer, dining philosophers, etc.).

Unit-VII [06] Max Marks:10
Deadlocks: modelling, characterization, prevention and avoidance, detection and recovery.

Unit-VIII [07] Max Marks:12
Memory management: with and without swapping, paging and segmentation, demand paging, virtual memory, page replacement algorithms, working set model, implementations from operating systems such as UNIX. Current Hardware support for paging: e.g., Pentium/ MIPS.
processor etc.

Unit-IX  [07]  Max Marks: 12
Secondary storage and Input/Output: device controllers and device drivers, disks, scheduling algorithms, file systems, directory structure, device controllers and device drivers, disks, disk space management, disk scheduling, NFS, RAID, other devices. operations on them, UNIX FS, UFS protection and security, NFS

Unit-X  [04]  Max Marks: 06
Protection and security: Illustrations of security model of UNIX and other OSs. Examples of attacks.

Unit-XI  [03]  Max Marks: 06
Epilogue: Pointers to advanced topics (distributed OS, multimedia OS, embedded OS, real-time OS, OS for multiprocessor machines).

All above topics shall be illustrated using UNIX as case-studies.

References:
Construction, Packages, The Class Path, Documentation Comments

Unit-III

Inheritance, Interfaces, Lambda Expressions, and Inner Classes: Classes, Super classes, and Subclasses, Object: The Cosmic Superclass, Generic Array Lists, Object Wrappers and Autoboxing, Methods with a Variable Number of Parameters, Enumeration, Classes, Reflection, Interfaces, Examples of Interfaces, Lambda Expressions, Inner Classes, Proxies

Max Marks: 16

Unit-IV


Max Marks: 16

Unit-V


Max Marks: 24

Unit-VI


Max Marks: 20

References:
Course Code: CS LAB-I LAB on Data Structures and Algorithms and JAVA programming Total Marks: 100

Data Structures and Algorithms
1. Implementation of programs based on the following
   - Arrays
   - Multidimensional Arrays, Matrices
   - Stacks, Polish Notation
   - Queues
   - Deques
   - Linear Linked List, Circular Linked List, Doubly Linked List
   - Polynomial Addition/Subtraction
2. Implementation of programs based on Trees
   - Binary Search Tree
   - In-order, Pre-order and Post-order Traversals
   - Heap Tree
   - Balanced Binary Tree (AVL)
   - B-Trees
3. Implementation of programs based on Graphs
   - Depth First Traversal
   - Breadth First Traversal
   - Obtaining Shortest Path (Dijkstra and Floyd-Warshall)
   - Minimum spanning tree (Kruskal and Prim)
4. Implementation of programs for Hash Table, Searching and Sorting techniques
   - Hash Table
   - Linear and Binary Search (using array)
   - Bubble sort
   - Selection sort
   - Insertion sort
   - Radix sort
   - Quick sort
   - Merge sort
   - Heap sort

JAVA programming
1. Write a program that demonstrates program structure of java.
2. Write a program that demonstrates string operations.
3. Write a program that demonstrates package creation and use in program.
4. Write a program that demonstrate inner class.
5. Write a program that demonstrates inheritance.
6. Write a program that demonstrates 2D shapes on frames.
7. Write a program that demonstrates text and fonts.
8. Write a program that demonstrates event handling for various types of events.
9. Write a program to illustrate use of various swing components.
10. Write a program that demonstrates use of dialog box.
11. Write a program to create own dialog box.
12. Write a program to create toolbar, menu & popup menu.
13. Write a program to implement file handleings.
14. Write a program that demonstrates Applet programming.
15. Write a program to implement generic programming.
16. Write a program that demonstrates JDBC on applet/application.
17. Write a program that demonstrates multithreading.

Course Code: CS LAB-II  
LAB on DBMS  
Total Marks: 100

1. Creating database tables and using data types.
   • Create table
   • Modify table
   • Drop table

2. Practical Based on Data Manipulation.
   • Adding data with Insert
   • Modify data with Update
   • Deleting records with Delete

3. Practical Based on Implementing the Constraints.
   • NULL and NOT NULL
   • Primary Key Constraint
   • Foreign Key Constraint
   • Unique Constraint
   • Check Constraint
   • Default Constraint

4. Practical for Retrieving Data Using following clauses.
   • Simple select clause
   • Accessing specific data with Where
   • Ordered By
   • Distinct
• Group By
5. Practical Based on Aggregate Functions.
   • AVG
   • COUNT
   • MAX
   • MIN
   • SUM
   • CUBE
6. Practical Based on implementing all String functions.
7. Practical Based on implementing Date and Time Functions.
8. Practical Based on implementing use of UNION, INTERSECTION, SET DIFFERENCE.
9. Implement Nested Queries & all types of JOIN operation.
10. Practical Based on performing different operations on a view.
11. Practical Based on implementing use of Procedures.
12. Practical Based on implementing use of Triggers
13. Practical Based on implementing Cursor.
14. ++++VB.NET, C#.NET, JAVA, D2K, etc.
15. Practical based on creating Data Reports.
16. Design entity relationship models for a business problem and develop a normalized database structure
Semester-II

Course Code: CS-201  Compiler Construction  Clock Hours: 60  
Total Marks: 100

Unit-I  [05]  Max Marks: 10
Compiler structure: analysis-synthesis model of compilation, various phases of a compiler, tool-based approach to compiler construction.

Unit-II  [06]  Max Marks: 15
Lexical analysis: Interface with input, parser and symbol table, token, lexeme and patterns, Difficulties in lexical analysis, Error reporting, Implementation, Regular definition, Transition diagrams, LEX.

Unit-III  [15]  Max Marks: 30
Syntax analysis: CFGs, ambiguity, associativity, precedence, top down parsing, recursive descent parsing, transformation on the grammars, predictive parsing, bottom up parsing, operator precedence grammars, LR parsers (SLR, LALR, LR), YACC.
Syntax directed definitions: inherited and synthesized attributes, dependency graph, evaluation order, bottom up and top down evaluation of attributes, L- and S-attributed definitions.
Type checking: type system, type expressions, structural and name equivalence of types, type conversion, overloaded functions and operators, polymorphic functions.

Unit-IV  [10]  Max Marks: 15
Run time system: storage organization, activation tree, activation record, parameter passing, symbol table, dynamic storage allocation.

Unit-V  [10]  Max Marks: 15

Unit-VI  [10]  Max Marks: 15
Code generation and instruction selection: issues, basic blocks and flow graphs, register allocation, code generation, dag representation of programs, code generation from dags, peep hole optimization, code generator generators, specifications of machine.

References:
2. Barret, Couch. Compiler Construction Theory and Practice: Computer Science series,
School of Computer Sciences, NMU, Jalgaon

Asian Student Ed, ISBN 978-0574213358

<table>
<thead>
<tr>
<th>Course Code: CS-202</th>
<th>Mathematical Foundations of Computer Science</th>
<th>Clock Hours: 60</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Unit-II</strong></td>
<td>Advance Counting Techniques: Recursive Relations, Solving Linear Recurrence Relations, Divide-and-Conquer Algorithms and Recurrence Relations, Generating Functions, Inclusion-Exclusion, Applications of Inclusion-Exclusion.</td>
<td>Max Marks: 20</td>
</tr>
<tr>
<td><strong>Unit-III</strong></td>
<td>Statistics: Population and sample, parameters and statistics: definition, types: Descriptive and Inferential, applications, Descriptive Statistics: Mean, median, mode and standard deviation, variance, Graphical statistics</td>
<td>Max Marks: 10</td>
</tr>
<tr>
<td><strong>Unit-V</strong></td>
<td>Stochastic Processes: Definitions and classifications of Stochastic Processes, discrete and continuous Markov models, Hidden Markov Models, Chapman-Kolmogorov equation</td>
<td>Max Marks: 15</td>
</tr>
</tbody>
</table>

**References:**

Course Code: CS-203 Artificial Intelligence Clock Hours: 60
Total Marks: 100

Unit-I [08] Max Marks:10
Introduction: Overview and Historical Perspective, Turing test, Physical Symbol Systems and the scope of Symbolic AI, Agents.

Unit-II [06] Max Marks:10
State Space Search: Depth First Search, Breadth First Search, DFID.

Unit-III [08] Max Marks:12
Heuristic Search: Best First Search, Hill Climbing, Beam Search, Tabu Search.

Unit-IV [08] Max Marks:15

Unit-V [08] Max Marks:12

Unit-VI [06] Max Marks:12
Game Playing: Minimax Algorithm, AlphaBeta Algorithm, SSS*.

Unit-VII [08] Max Marks:14
Planning and Constraint Satisfaction: Domains, Forward and Backward Search, Goal Stack Planning, Plan Space Planning, Graphplan, Constraint Propagation.

Unit-VIII [08] Max Marks:15
Logic and Inferences: Propositional Logic, First Order Logic, Soundness and Completeness, Forward and Backward chaining.
References:

Course Code: CS-204 Design and Analysis of Algorithms Clock Hours: 60 Total Marks: 100

Unit-I
Tree And Graph Representations, Binary Trees Basics, Heaps And Heap Sort, Sets And Disjoint Set Union And Find.

Unit-II
Divide and Conquer: General Method, Binary Search, Finding Maximum and Minimum, Merge Sort, Quick Sort, Strassen’s Matrix Multiplication.

Unit-III

Unit-IV
Dynamic Programming: General Method, All-Pair Shortest Path, Matrix Chain Multiplication, Longest Common Sub Sequence, 0/1knapsack, Flow Shop Scheduling.

Unit-V
Basic Search and Traversal Techniques: Breadth First Search and Traversal, Depth First Search And Traversal, Spanning Trees.

Unit-VI
Backtracking: General Method, Constrains, 8-Queens Problem Graph Coloring.
References:

Course Code: CS-205  Python Programming  Clock Hours: 60  Total Marks: 100

Unit-I  [12]  Max Marks:20

Unit-II  [08]  Max Marks:15
Strings, A Collection Data Type, Operations on Strings, Index Operator: Working with the Characters of a String, String Methods, Length, The Slice Operator, String Comparison, Lists, List Values, List Length, Accessing Elements, List Membership, Concatenation and Repetition, List Slices, Lists are Mutable, List Deletion, Objects and References, Aliasing, Cloning Lists, Repetition and References, List Methods, Append versus Concatenate Lists and for loops, Using Lists as Parameters, Nested Lists, Strings and Lists, List Type Conversion Function, Tuples, Tuple operators and built-in functions, Tuples and Mutability, Tuple Assignment, Tuples as Return Values

Unit-IV

Max Marks: 20


Unit-V

Max Marks: 25


References:

Course Code: CS-LAB-III   LAB on Design and Analysis of Algorithms (DAA)   Total Marks: 100

OS: Windows/Linux,  Programming Language: C++/Java/C#

1. Write a program for creating max./min. heap using
   - INSERT
   - ADJUST/HEAPIFY
2. Write a program to implement union and find operation.
3. Write a program to find minimum and maximum form a given array.
4. Write a program for searching element form given array using binary search for
   n=1000, 2000, 3000 find exact time of execution.
5. Write a program for sorting given array in ascending/descending order with
   n=1000, 2000, 3000 find exact time of execution using
   - Heap sort
   - Merge sort
   - Quick sort
6. Write a program for matrix multiplication using Strassen’s matrix multiplication.
7. Write a program to find solution of Knapsack instant.
8. Write a program to find shortest path using single source shortest path.
9. Write a program to find Minimum-Cost Spanning Trees (Prim’s & Kruskal’s Algorithm).
10. Write a program to find shortest path using all pair path.
11. Write a program to find longest common subsequence.
12. Write a program to implement breadth first and depth first search.
13. Write a program to implement breadth first and depth first traversal.
14. Write a program to find all solutions for 8-queen problem using backtracking.
15. Write a program for creating max./min. heap using
    - INSERT
    - ADJUST/HEAPIFY
16. Write a program to implement union and find operation.
17. Write a program to find minimum and maximum form a given array.
18. Write a program for searching element form given array using binary search for
    n=1000, 2000, 3000 find exact time of execution.
19. Write a program for sorting given array in ascending/descending order with
    n=1000, 2000, 3000 find exact time of execution using
    - Heap sort
    - Merge sort
    - Quick sort
20. Write a program for matrix multiplication using Strassen’s matrix multiplication.
21. Write a program to find solution of Knapsack instant.
22. Write a program to find shortest path using single source shortest path.
23. Write a program to find Minimum-Cost Spanning Trees (Prim’s & Kruskal’s Algorithm).
24. Write a program to find shortest path using all pair path.
25. Write a program to find longest common subsequence.
26. Write a program to implement breadth first and depth first search.
27. Write a program to implement breadth first and depth first traversal.
28. Write a program to find all solutions for 8-queen problem using backtracking.

Course Code: CS-LAB-IV  LAB on Python Programming  Total Marks: 100

1. Develop programs to understand the control structures of python
2. Develop programs to learn different types of structures (list, dictionary, tuples) in python
3. Develop programs to learn concept of functions scoping, recursion and list mutability.
4. Develop programs to understand object oriented programming using python.
5. Develop programs for data structure algorithms using python – searching, sorting and hash tables.
6. Develop programs to learn regular expressions using python.
7. Develop programs to learn GUI programming using Tkinter.
8. Demonstrate the concept of exception handling using try/except/else Statement, Unified try/except/finally, try/finally Statement, raise Statement, assert Statement, catch multiple specific exceptions
9. Demonstrate the concept of String-Based Exceptions, Class-Based Exceptions and Nesting Exception handlers.
10. Demonstrate implementation of the Anonymous Function Lambda.
11. Demonstrate implementation Mapping Functions over Sequences.
12. Demonstrate implementation functional programming tools such as filter and reduce
13. Demonstrate the Module Creation, Module usage, Module Namespaces, Reloading Modules, Module Packages, Data Hiding in Modules.
14. Demonstrate Mixed Usage Modes of modules, Changing the Module Search Path, The import as Extension, Relative Import Syntax, Module Design Concepts
Semester-III

Course Code: CS-301  Machine Learning  Clock Hours: 60
Total Marks: 100

Unit-I [08] Max Marks:10
Introduction: Basic definitions, types of learning, hypothesis space and inductive bias, evaluation, cross-validation

Unit-II [08] Max Marks:15
Linear regression, Decision trees, overfitting

Unit-III [09] Max Marks:15
Instance based learning, Feature reduction, Collaborative filtering based recommendation

Unit-IV [08] Max Marks:15
Probability and Bayes learning

Unit-V [09] Max Marks:15
Logistic Regression, Support Vector Machine, Kernel function and Kernel SVM

Unit-VI [09] Max Marks:15
Neural network: Perceptron, multilayer network, backpropagation, introduction to deep neural network

Unit-VII [09] Max Marks:15
Clustering: k-means, adaptive hierarchical clustering, Gaussian mixture model

References:
Course Code: CS-302  Natural Language Processing  Clock Hours: 60  Total Marks: 100

Unit-I [08] Max Marks: 12
Introduction to NLP, brief history, NLP applications: Speech to Text (STT), Text to Speech (TTS), Story Understanding, NL Generation, QA system, Machine Translation, Text Summarization, Text classification, Sentiment Analysis, Grammar/Spell Checkers etc., challenges/Open Problems, NLP abstraction levels, Natural Language (NL) Characteristics and NL computing approaches/techniques and steps, NL tasks: Segmentation, Chunking, tagging, NER, Parsing, Word Sense Disambiguation, NL Generation, Web 2.0 Applications: Sentiment Analysis; Text Entailment; Cross Lingual Information Retrieval (CLIR).

Unit-II [12] Max Marks: 16
Text Processing Challenges, Overview of Language Scripts and their representation on Machines using Character Sets, Language, Corpus and Application Dependence issues, Segmentation: word level (Tokenization), Sentence level. Regular Expression and Automata Morphology, Types, Survey of English and Indian Languages Morphology, Morphological parsing FSA and FST, Porter stemmer, Rule based and Paradigm based Morphology, Human Morphological Processing, Machine Learning approaches

Unit-III [12] Max Marks: 18
Word Classes ad Part-of-Speech tagging (POS), survey of POS tagsets, Rule based approaches (ENGTOWL), Stochastic approaches (Probabilistic, N-gram and HMM), TBL morphology, unknown word handling, evaluation metrics: Precision/Recall/F-measure, error analysis

Unit-IV [15] Max Marks: 22
NL parsing basics, approaches: TopDown, BottomUp, Overview of Grammar Formalisms: constituency and dependency school, Grammar notations CFG, LFG, PCFG, LTAG, Feature-Unification, overview of English CFG, Indian Language Parsing in Paninian Karaka Theory, CFG parsing using Earley’s and CYK algorithms, Probabilistic parsing

Concepts and issues in NL, Theories and approaches for Semantic Analysis, Meaning Representation, word similarity, Lexical Semantics, word senses and relationships, WordNet (English and IndoWordnet), Word Sense Disambiguation: Lesk Algorithm Walker’s algorithm, Coreferences Resolution: Anaphora, Cataphora

References:

1. Indurkhya, N., & Damerau, F. J. (Eds.). (2010). Handbook of Natural Language


Auxiliary Resources:

1. Web Links
   1. https://see.stanford.edu/Course/CS224N

2. Video Links
   2. https://www.youtube.com/playlist?list=PL6397E4B26D00A269

Course Code: CS-303 Computer Graphics and Digital Image Processing

Clock Hours: 60 Total Marks: 100


Unit-II Two-Dimensional Transformation: Matrix and transformation, 2D Basic transformation, Homogeneous coordinates, Translation, Scaling and Rotation of straight line or polygon, Combined Transformation, Rotation about an arbitrary point/line, Reflection and Shearing
Transformation, Viewing Transformation, Clipping, Cohen-Sutherland line clipping.

Unit-III  
**Three-Dimensional Transformation:** Introduction, Matrix representation of 3D Transformation, 3D Translation, Scaling, Rotation, Composition of 3D Transformation, Projection, Orthographic, Isometric, Oblique Projection, Perspective Projection, One-Two-Three point perspective Projection.

Unit-IV  

Unit-V  
**Image Enhancement:** Background, Some Basic Gray Level Transformations, Histogram Processing, Enhancement Using Arithmetic/Logic Operations, Basics of Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters, Combining Spatial Enhancement Methods, Introduction to the Fourier Transform and the Frequency Domain, Smoothing Frequency-Domain Filters, Sharpening Frequency Domain Filters, Homomorphic Filtering

Unit-VI  

Unit VII  
**Morphological Image Processing & Segmentation:** Detection of Discontinuities, Edge linking & Boundary Detection, Thresholding, Region based segmentation Laplacian of Gaussian, Derivative of Gaussian, Canny Edge Detection, Morphological operation: Dilation erosion, Opening & Closing, Basic Morphological Algorithm, Image representation schemes.

Unit VIII  
**MATLAB Image processing toolbox:** Introduction to MATLAB, Matrix Operations, Introduction to Image Processing Tool Box, Image Read & Write, Filters (spatial and frequency domain), Image Restoration and Reconstruction, Morphological Operations, Edge
Detection and linking, Segmentation.

References:

Course Code: CS-304 Software Engineering Clock Hours: 60 Total Marks: 100


Unit-III Quality Assurance and Change Management: Elements of SQA, SQA Tasks, Goal and Metrics, Formal approaches to SQA, Software Reliability, ISO 9000 Quality standards, SQA Plan. Software Configuration Management, SCM Repository, SCM process
Unit-IV


Unit-V


**Architectural and Component Level Design:** Software Architectures, Architectural Genres, Architectural styles, Architectural design, Accessing alternatives Architectural design, Architectural mapping using dataflow. Introduction to component, Designing class based component, Conducting component level design, Designing traditional component, component based development.

Unit-VI

[10] Max Marks: 20

**Software Testing:** Strategic approach to software testing, Test strategies for conventional software, Validation Testing, System testing, Software testing fundamentals, Internal and external view of testing, White box testing, Basic path testing, Control structure testing, Black box testing, model based testing, Testing for specialized Environment, Architectures and applications.

**References:**


**Course Code:** CS-305

**Web Application Development Technology**

**Clock Hours:** 60

**Total Marks:** 100

Unit-I

[10] Max Marks: 15

Pipeline, HTTP Application, HTTP Request, HTTP Response classes, HTTP Modules and HTTP Handlers, State Management, Role of Global.asax, Application configuration using web.config file

**Unit-II** [15]  Max Marks:25
ASP.NET Control hierarchy, HTML Server Controls, Web Server Controls, User and Server controls, Validation Controls, List bound controls: dropdown lists, list boxes, Repeater, DataList, Data Grid, DataGridView, DataGridView, FormsView controls, Data binding to List Bound Controls, Templating and Styling of ASP.NET server controls

**Unit-III** [20]  Max Marks:25

**Unit-IV** [15]  Max Marks:25
Database technology: ADO.NET, Anatomy/architecture of ADO.NET, working with Connection, Command, Data Adaptor, DataReader, DataSet, DataTable objects, Editing data in Data Tables, concurrency control. Introduction to MVC, Data Reports

**References:**

**Auxiliary Resources:**
Website URLs
- https://www.asp.net/
- http://asp.net-tutorials.com/

Video Links
Course Code: CS LAB-V | LAB on Web Application Development Technology | Total Marks: 100

1. Demonstrate followings in IIS:
   a. Creation of Virtual Directory, Home directory, Home page, hosting of website
2. Demonstrate Page Life Cycle of ASP.NET. Use important page events for your demonstration.
3. Write VB.Net/C# console applications to demonstrate: OO concepts: polymorphism, encapsulation, inheritance, interface inheritance, abstract classes/methods, overloading, overriding, collection classes, properties
4. Demonstrate concept of postback and viewstate using web form server controls of ASP.NET
5. Demonstrate various Web form server controls using sample data entry screen form for registering for a service on website. Also use validation controls to validate input data.
7. Demonstrate Databinding using Hashtable, ArrayList, DataTable data sources.
8. Demonstrate Repeater control with the help of various templates.
10. Demonstrate editing process in DataGrid and DataList controls. Make use of necessary templates for proper visual appearance.
11. Demonstrate State Management features of ASP.NET using sample shopping cart application.
12. Create sample website for demonstrating use of Profiles/Themes using skin files.
13. Demonstrate Master Pages and website navigation controls(sitemap path, treeview, menu) using SiteMap file.
15. Demonstrate Authorization/Authentication using Login controls and Roles/Membership/AccessRules
16. Demonstrate creation of simple/complex DataReader/DataSet Objects.
17. Demonstrate editing in DataTable objects.
18. Demonstrate Web Service hosting, access in ASP.NET
Course Code: CS LAB-VI  
LAB on Computer Graphics and Digital Image Processing  
Total Marks: 100

1. Line drawing algorithm (DDA and Bresenham’s Line Algorithm)
2. Circle drawing algorithm
3. Ellipse drawing algorithm
4. Polygon filling algorithm
5. Windowing and clipping algorithm (Point, line and polygon clipping)
6. Composite 2-D transformation, (rotation, scaling & reflection)
7. 3-D geometric transformation (rotation, scaling & reflection)
8. Introduction to Image Processing Toolbox
9. Read an 8 bit image and then apply different image enhancement techniques:
   - Brightness improvement
   - Brightness reduction
   - Thresholding
   - Negative of an image
   - Log transformation
   - Power Law transformation.
10. Implement different interpolation techniques using MATLAB/ SciLab
11. Read an image, plot its histogram then do histogram equalization. Comment about the result.
12. Read an image and apply
   - Implement Gray level slicing (intensity level slicing) in to read cameraman image.
   - Read an 8 bit image and to see the effect of each bit on the image.
   - Read an image and to extract 8 different planes i.e. ‘bit plane slicing.”
13. Implement various Smoothing spatial filter.
14. Read an image and apply
   - Gaussian 3x3 mask for burring
   - High pass filter mask with different masks
   - Laplacian operator with centre value positive and negative
   - High boost filtering.
15. Write a program to implement various low pass filters and high pass filter in frequency domain.
16. Write a program for erosion and dilation, opening & closing using inbuilt and without inbuilt function.
17. Implement and study the effect of Different Mask (Sobel, Prewitt and Roberts)
18. Implement various noise models and their Histogram
19. Implement inverse filter and wiener filter over image and comment on them
Semester- IV

CS-401 Full Time Industrial Training

Twelve credits shall be awarded to the Industrial Training/Project course, which will commence in the IVth Semester and the final work and report will be completed at the end of IVth Semester of M. Sc. (Computer Science). The student is expected to work on software development project. The project work should have coding part. Student will have to submit the bound project report in university prescribed format at the end of the semester. Student will have to appear for Project Viva-voce and the marks and the credits will be allotted at the end of IVth semester of M. Sc. (Computer Science).