

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS I Year II Semester

Data Structures & Algorithms

Prerequisites

- A course on “Computer Programming & Data Structures”

Course Objectives:

- Introduces the notations for analysis of the performance of algorithms
- Introduces a variety of data structures such as hash tables, disjoint sets and Priority Queue
- Describes major algorithmic techniques (divide-and-conquer, backtracking, dynamic programming and greedy) and mention problems for which each technique is appropriate;
- Introduces sorting, searching and pattern matching algorithms

Course Outcomes:

- Ability to analyze the performance of algorithms
- Ability to select the data structures that efficiently model the information in a problem
- Ability to choose appropriate data structures and algorithm design methods for a specified application
- Ability to understand how the choice of data structures and the algorithm design methods impact the performance of programs
- Design programs using a variety of data structures, including hash tables, disjoint sets, trees and graphs
- Implement and know the application of algorithms for sorting and pattern matching

UNIT - I

Introduction: Algorithm, Pseudo code for expressing algorithms, Performance Analysis-Space complexity, Time complexity, Asymptotic Notations, Substitution method, Master's theorem

Divide and conquer: General method, applications-Binary search, Quick sort, Merge sort, Strassen's matrix multiplication

UNIT - II

Disjoint Sets: Disjoint set operations, union and find algorithms

Greedy method: General method, applications-Job sequencing with deadlines, 0/1 knapsack problem, Minimum cost spanning trees, Single source shortest path problem.

Priority Queues – Definition, Realizing a Priority Queue using Heaps, operations of priority queue

UNIT – III

Dynamic Programming: General method, applications- Optimal binary search trees, 0/1 knapsack problem, All pairs shortest path problem, Travelling sales person problem, Reliability design.

Backtracking: General method, applications-n-queens problem, sum of subsets problem, graph coloring.

UNIT – IV

Review of basic data structures: The list, Stack, Queue

linear list representation, skip list representation, operations - insertion, deletion and searching.

Hash table representation: hash functions, collision resolution-separate chaining, open addressing-linear probing, quadratic probing, double hashing, rehashing, extendible hashing, comparison of hashing and skip lists.

Searching: Linear Search and Binary Search

Sortings: Bubble sort, Selection sort, Insertion sort, Radix sort, Heap sort

UNIT – V

Trees: Definition, Definition, Types of trees, Binary Trees, Binary Tree Traversal Methods, Binary search tree operations, operations of AVL tree, B-Tree, Red-Black tree, Splay tree

Graphs: Definition, Representation of graphs, Graphs Traversal Methods.

String Matching algorithms: Brute Force algorithm, Boyer Moore algorithm, Knuth- Morris-Pratt algorithm

TEXT BOOKS:

1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharam, Universities Press.
2. Fundamentals of Data Structures, Ellis Horowitz, Satraj Sahni and Rajasekharam, Universities Press.

REFERENCES:

1. Introduction to Algorithms, second edition, T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, PHI Pvt. Ltd./ Pearson Education.
2. Data structures using C and C++, Langsam, Augenstein and Tanenbaum, PHI.
3. Data structures: A Pseudocode Approach with C, 2 nd edition, R. F. Gilberg and B.A. Forouzan, Cengage Learning.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS I Year II Semester

OBJECT ORIENTED PROGRAMMING THROUGH C++

UNIT - I

C++ Basics: Structure of a C++ program, Data types, Declaration of variables, Expressions, Operators, Operator Precedence, Evaluation of expressions, Type conversions, Pointers, Arrays, Pointers and Arrays, Strings, Structures, References.

Flow control statement- if, switch, while, for, do, break, continue, goto statements.

Functions - Scope of variables, Parameter passing, Default arguments, inline functions, Recursive functions, Pointers to functions.

Dynamic memory allocation and deallocation operators-new and delete, Preprocessor directives.

UNIT - II

Different paradigms for problem solving, need for OOP, differences between OOP and Procedure oriented programming, Abstraction, Overview of OOP principles, Encapsulation, Inheritance and Polymorphism.

C++ Classes And Data Abstraction: Class definition, Class structure, Class objects, Class scope, this pointer, Friends to a class, Static class members, Constant member functions, Constructors and Destructors, Dynamic creation and destruction of objects, Data abstraction, ADT and information hiding.

UNIT- III

Inheritance: Defining a class hierarchy, Different forms of inheritance, Defining the Base and Derived classes, Access to the base class members, Base and Derived class construction, Destructors, Virtual base class.

Virtual Functions And Polymorphism: Static and Dynamic bindings, Base and Derived class virtual functions, Dynamic binding through virtual functions, Virtual function call mechanism, Pure virtual functions, Abstract classes, Implications of polymorphic use of classes, Virtual destructors.

UNIT - IV

C++ I/O: I/O using C functions, Stream classes hierarchy, Stream I/O, File streams and String streams, Overloading << and >> operators, Error handling during file operations, Formatted I/O.

UNIT - V

Exception Handling: Benefits of exception handling, Throwing an exception, The try block, Catching an exception, Exception objects, Exception specifications, Stack unwinding, Rethrowing an exception, Catching all exceptions, Design issues in exception handling.

Text Books:

1. Problem solving with C++: The Object of Programming, 4th Edition, Walter Savitch, Pearson Education.
2. C++, the Complete Reference, 4th Edition, Herbert Schildt, TMH.

REFERENCES:

1. C++ Primer, 3rd Edition, S. B. Lippman and J. Lajoie, Pearson Education.
2. The C++ Programming Language, 3rd Edition, B. Stroustrup, Pearson Education.
3. OOP in C++, 3rd Edition, T. Gaddis, J. Walters and G. Muganda, Wiley DreamTech Press.
4. Object Oriented Programming in C++, 3rd Edition, R. Lafore, Galigotia Publications pvt ltd.
5. Computer Science, a Structured Programming Approach Using C++, B. A. Forouzan and R. F. Gilberg, Thomson.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS I Year II Semester

SCRIPTING LANGUAGES

Prerequisites:

- A course on “Computer Programming and Data Structures”
- A course on “Object Oriented Programming Concepts”

Course Objectives:

- This course provides an introduction to the script programming paradigm
- Introduces scripting languages such as Perl, PHP and Python.
- Learning TCL

Course Outcomes:

- Comprehend the differences between typical scripting languages and typical system and application programming languages.
- Gain knowledge of the strengths and weakness of Perl, PHP TCL and Python; and select an appropriate language for solving a given problem

UNIT – I

Introduction to PERL and Scripting, Scripts and Programs, Origin of Scripting , Scripting Today, Characteristics of Scripting Languages, Uses for Scripting Languages, Web Scripting, and the universe of Scripting Languages. PERL- Names and Values, Variables, Scalar Expressions, Control Structures, arrays, list, hashes, strings, pattern and regular expressions, subroutines.

UNIT – II Advanced perl,

Finer points of looping, pack and unpack, filesystem, eval, data structures, packages, modules, objects, interfacing to the operating system, Creating Internet ware applications, Dirty Hands Internet Programming, security Issues.

UNIT – III PHP Basics

Features, Embedding PHP Code in your Web pages, Outputting the data to the browser, Datatypes, Variables, Constants, expressions, string interpolation, control structures . Function, Creating a Function, Function Libraries, Arrays, strings and Regular Expressions.

PHP and Web Forms, Files, PHP Authentication and Methodologies -Hard Coded, File Based, Database Based, IP Based, Login Administration, Uploading Files with PHP, Sending Email using PHP, PHP Encryption Functions, the Mcrypt package, Building Web sites for the World.

UNIT - IV TCL

TCL Structure, syntax, Variables and Data in TCL, Control Flow, Data Structures, input/output, procedures , strings , patterns, files, Advance TCL- eval, source, exec and uplevel commands, Name spaces, trapping errors, event driven programs, making applications internet aware, Nuts and Bolts Internet Programming, Security Issues, C Interface.

Tk

Tk-Visual Tool Kits, Fundamental Concepts of Tk, Tk by example, Events and Binding , Perl-Tk.

UNIT – V Python

Introduction to Python language, python-syntax, statements, functions, Built-in-functions and Methods, Modules in python, Exception Handling.

Integrated Web Applications in Python – Building Small, Efficient Python Web Systems ,Web Application Framework.

TEXT BOOKS:

1. The World of Scripting Languages, David Barron, Wiley Publications.
2. Python Web Programming, Steve Holden and David Beazley, New Riders Publications.
3. Beginning PHP and MySQL , 3rd Edition , Jason Gilmore, Apress Publications (Dream tech.).

REFERENCE BOOKS:

1. Open Source Web Development with LAMP using Linux Apache, MySQL, Perl and PHP, J. Lee and B. Ware (Addison Wesley) Pearson Education.
2. Programming Python,M.Lutz,SPD.
3. PHP 6 Fast and Easy Web Development, Julie Meloni and Matt Telles, Cengage Learning Publications.
4. PHP 5.1,I. Bayross and S. Shah, The X Team, SPD.
5. Core Python Programming, Chun, Pearson Education.
6. Guide to Programming with Python, M. Dawson, Cengage Learning.
7. Perl by Example, E. Quigley, Pearson Education.
8. Programming Perl, Larry Wall, T. Christiansen and J. Orwant, O'Reilly, SPD.
9. Tcl and the Tk Tool kit, Ousterhout, Pearson Education.
10. PHP and MySQL by Example, E. Quigley, Prentice Hall (Pearson).
11. Perl Power, J.P. Flynt, Cengage Learning.
12. PHP Programming solutions, V. Vaswani, TMH.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS I Year II Semester

OPERATIONS RESEARCH

Prerequisites

- A course on "Mathematics"

Course Objectives

- This course explains various Real time problems and Operation Research Techniques to address those problems.
- To study Linear Programming, dynamic programming Operation Research Techniques etc.
- To understand Theory of games.

Course Outcomes

- Gain the knowledge of Operation Research Techniques
- Get the skill to apply Operation Research Techniques to address the real time problems .

UNIT - I

DEVELOPMENT DEFINITION: Characteristics and phases scientific Method, Types of models, General methods for solving. Operations Research models.

ALLOCATION: Introduction, Linear programming Formulation, Graphical solution, Simplex method, artificial variable technique, Duality principle.

TRANSPORTATION PROBLEM: Formulation, optimal solution, un-balanced transportation problem, Degeneracy. Assignment problem: formulation optimal solution, variations. 1.a non-square (mxn) Matrix, Restrictions.

UNIT - II

EEQUE CEING: Introduction, optimal solution for processing each of n-jobs through three machines, travelling salesman problem i.e., shortest acyclic route models.

REPLACEMENT: Introduction, replacement of items that deteriorate when money value is not counted and counted, replacement items that fail completely i.e., group replacements.

UNIT - III

WAITING LINES: Introduction, single channel, poisson arrivals, exponential service times, unrestricted queue, with infinite population and finite population models, single channel, poisson arrivals, exponential service times with infinite population and restricted queue, multi channel, poisson arrivals, exponential service times with infinite population and unrestricted queue.

UNIT - IV

INVENTORY: Introduction, single item deterministic models, production is instantaneous or at a constant rate, shortages are allowed or not allowed and withdrawals from stock is continuous, purchase inventory model with one price break, shortages are not allowed, Instantaneous production demand, production or purchase cost is relevant, stochastic models, demand may be discrete or variable or instantaneous production, instantaneous demand and no setup cost.

UNIT - V

THEORY OF GAMES: Introduction, Minimax (maximum) criterion and optimal strategy, solution of games with saddle points, rectangular games without saddle points.

DYNAMIC PROGRAMMING: Introduction, Billman's Principal of optimality, solution of problems with finite number of stages.

TEXT BOOKS:

1. S. D. Sharma : Operations Research
kedar Nath Ramnath, 1972
2. P. K. Gupta & D. S. Hira : Operations Research
3. R. D. Asrhedkar & R. V. Kulkarni : Operations Research.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS I Year II Semester

SOFTWARE ENGINEERING

Prerequisites

- A course on “Computer Programming and Data Structures”
- A course on “Object Oriented Programming Through Java”

Course Objectives

- The aim of the course is to provide an understanding of the working knowledge of the techniques for estimation, design, testing and quality management of large software development projects.
- Topics include process models, software requirements, software design, software testing, software process/product metrics, risk management, quality management and UML diagrams

Course Outcomes

- Ability to translate end-user requirements into system and software requirements, using e.g. UML, and structure the requirements in a Software Requirements Document (SRD).
- Identify and apply appropriate software architectures and patterns to carry out high level design of a system and be able to critically compare alternative choices.
- Will have experience and/or awareness of testing problems and will be able to develop a simple testing report

UNIT - I:

Introduction to Software Engineering: The evolving role of software, changing nature of software, software myths.

A Generic view of process: Software engineering- a layered technology, a process framework, the capability maturity model integration (CMMI), process patterns, process assessment, personal and team process models.

Process models: The waterfall model, incremental process models, evolutionary process models, the unified process.

UNIT- II:

Software Requirements: Functional and non-functional requirements, user requirements, system requirements, interface specification, the software requirements document.

Requirements engineering process: Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management.

System models: Context models, behavioral models, data models, object models, structured methods.

UNIT- III:

Design Engineering: Design process and design quality, design concepts, the design model.

Creating an architectural design: software architecture, data design, architectural styles and patterns, architectural design, conceptual model of UML, basic structural modeling, class diagrams, sequence diagrams, collaboration diagrams, use case diagrams, component diagrams.

UNIT- IV:

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, black-box and white-box testing, validation testing, system testing, the art of debugging.

Product metrics: Software quality, metrics for analysis model, metrics for design model, metrics for source code, metrics for testing, metrics for maintenance.

UNIT - V:

Metrics for Process and Products: Software measurement, metrics for software quality.

Risk management: Reactive Vs proactive risk strategies, software risks, risk identification, risk projection, risk refinement, RMMM, RMMM plan.

Quality Management: Quality concepts, software quality assurance, software reviews, formal technical reviews, statistical software quality assurance, software reliability, the ISO 9000 quality standards.

TEXT BOOKS:

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition, Mc Graw Hill International Edition.
2. Software Engineering- Sommerville, 7th edition, Pearson Education.
3. The unified modeling language user guide Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education.

REFERENCES:

1. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiley.
2. Software Engineering principles and practice - Waman S Jawadekar, The Mc Graw-Hill Companies.
3. Fundamentals of object oriented design using UML Meiler page-Jones: Pearson Education.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS I Year II Semester

Data Structures Lab Using C++

Prerequisites

- A course on “Computer Programming & Data Structures” and “Advanced Data Base Engineering”.

Course Objectives

- To get practical exposure on Advanced Data Structures like AVL Trees, Red-Black trees etc.,
- implementation of data structures such as trees and graphs,
- Programming of sorting and pattern matching algorithms

Course Outcomes

- Ability to select the data structures that efficiently model the information in a problem.
- Ability to assess efficiency trade-offs among different data structure implementations or combinations.
- Implement and know the application of algorithms for sorting and pattern matching.
- Design programs using tree structures, including Optimal binary search tree, All pairs shortest path problem, AVL tree, Red-Black trees.

1. Write a program to implement the following sorting techniques
 - a. Merge sort
 - b. Quick sort
 - c. Bubble sort
 - d. Selection sort
 - e. Insertion sort
 - f. Radix sort
 - g. Heap sort
2. Write a program to implement the single source shortest path problem
3. Write a program to implement the Optimal binary search tree
4. Write a program to implement the All pairs shortest path problem
5. Write a program to implement the n-queens problem
6. Write a program to implement the following searching techniques
 - a. Linear Search
 - b. Binary Search
7. Write a program to implement the operations of Binary search tree
8. Write a program to implement the tree traversal methods
9. Write a program to implement the graph traversal methods
10. Write a program to implement the operations of AVL tree
11. Write a program to implement the operations of Red-Black tree
12. Write a program to implement the following Pattern matching algorithms
 - a. Brute Force algorithm
 - b. Boyer Moore algorithm
 - c. Knuth- Morris-Pratt algorithm

Pythagorean Theorem that in a right triangle, the square of one side equals the sum of the squares of the other two sides).

14. Write a python program to define a module to find Fibonacci Numbers and import the module to another program.
15. Write a python program to define a module and import a specific function in that module to another program.
16. Write a script named **copyfile.py**. This script should prompt the user for the names of two text files. The contents of the first file should be input and written to the second file.
17. Write a program that inputs a text file. The program should print all of the unique words in the file in alphabetical order.
18. Write a Python class to convert an integer to a roman numeral.
19. Write a Python class to implement $\text{pow}(x, n)$
20. Write a Python class to reverse a string word by word.

**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
MASTER OF COMPUTER APPLICATIONS I Year II Semester**

English Language Communication Skills Lab

Prerequisite: NIL

Course Objectives

- To facilitate computer-aided multi-media instruction enabling individualized and independent language learning
- To sensitise the students to the nuances of English speech sounds, word accent, intonation and rhythm
- To bring about a consistent accent and intelligibility in their pronunciation of English by providing an opportunity for practice in speaking
- To improve the fluency in spoken English and neutralize mother tongue influence
- To train students to use language appropriately for interviews, group discussion and public speaking

Learning Outcomes

- Better Understanding of nuances of language through audio- visual experience and group activities
- Neutralization of accent for intelligibility
- Speaking with clarity and confidence thereby enhancing employability skills of the students

Syllabus: English Language Communication Skills Lab shall have two parts:

- a. **Computer Assisted Language Learning (CALL) Lab**
- b. **Interactive Communication Skills (ICS) Lab**

The following course content is prescribed for the **English Language Communication Skills Lab**

Exercise – I

CALL Lab: Introduction to Phonetics – Speech Sounds – Vowels and Consonants

ICS Lab: Ice-Breaking activity and JAM session

Articles, Prepositions, Word formation- Prefixes & Suffixes, Synonyms & Antonyms

Exercise – II

CALL Lab: Structure of Syllables - Past Tense Marker and Plural Marker – Weak Forms and Strong Forms - Consonant Clusters.

ICS Lab: Situational Dialogues – Role-Play- Expressions in Various Situations – Self-introduction and Introducing Others – Greetings – Apologies – Requests – Social and Professional Etiquette - Telephone Etiquette.

Concord (Subject in agreement with verb) and Words often misspelt- confused/misused

Exercise - III

CALL Lab: Minimal Pairs- Word accent and Stress Shifts- Listening Comprehension.

ICS Lab: Descriptions- Narrations- Giving Directions and guidelines.

Sequence of Tenses, Question Tags and One word substitutes.

Exercise – IV

CALL Lab: Intonation and Common errors in Pronunciation.

ICS Lab: Extempore- Public Speaking

Active and Passive Voice, –Common Errors in English, Idioms and Phrases

Exercise – V

CALL Lab: Neutralization of Mother Tongue Influence and Conversation Practice

ICS Lab: Information Transfer- Oral Presentation Skills

Reading Comprehension and Job Application with Resume preparation.

Minimum Requirement of infrastructural facilities for ELCS Lab:

1. Computer Assisted Language Learning (CALL) Lab:

The Computer aided Language Lab for 40 students with 40 systems, one master console, LAN facility and English language software for self- study by learners.

System Requirement (Hardware component):

Computer network with Lan with minimum 60 multimedia systems with the following specifications:

- i) P – IV Processor
 - a) Speed – 2.8 GHZ
 - b) RAM – 512 MB Minimum
 - c) Hard Disk – 80 GB
- ii) Headphones of High quality

2. Interactive Communication Skills (ICS) Lab :

The Interactive Communication Skills Lab: A Spacious room with movable chairs and audio-visual aids with a Public Address System, a T. V., a digital stereo –audio & video system and camcorder etc.

Suggested Software:

1. **Cambridge Advanced Learners' English Dictionary with CD.**
2. **Grammar Made Easy by Darling Kindersley**
3. **Punctuation Made Easy by Darling Kindersley**
4. Clarity Pronunciation Power – Part I
5. Clarity Pronunciation Power – part II
6. **Oxford Advanced Learner's Compass, 8th Edition**
7. **DELTA's key to the Next Generation TOEFL Test: Advanced Skill Practice.**
8. Lingua TOEFL CBT Insider, by Dreamtech
9. TOEFL & GRE (KAPLAN, AARCO & BARRONS, USA, Cracking GRE by CLIFFS)
10. **English in Mind (Series 1-4), Herbert Puchta and Jeff Stranks with Meredith Levy, Cambridge**
11. **English Pronunciation in Use** (Elementary, Intermediate, Advanced) Cambridge University Press
12. Raman, M & Sharma, S. 2011. Technical Communication, OUP
13. Sanjay Kumar & Pushp Lata. 2011. Communication Skills, OUP

SUGGESTED READING:

1. Rama Krishna Rao, A. *et al. English Language Communication Skills – A Reader cum Lab Manual Course Content and Practice.* Chennai: Anuradha Publishers
2. Suresh Kumar, E. & Sreehari, P. 2009. *A Handbook for English Language Laboratories.* New Delhi: Foundation
3. *Speaking English Effectively 2nd Edition* by Krishna Mohan and N. P. Singh, 2011. Macmillan Publishers India Ltd. Delhi.
4. Sasi Kumar, V & Dhamija, P.V. *How to Prepare for Group Discussion and Interviews.* Tata McGraw Hill
5. Spoken English: A Manual of Speech and Phonetics by R. K. Bansal & J. B. Harrison. 2013. Orient Blackswan. Hyderabad.
6. *English Pronunciation in Use. (Elementary, Intermediate & Advance).* Cambridge: CUP

7. [Chris Redston](#), [Gillie Cunningham](#), Jan Bell. *Face to Face* (2nd Edition). Cambridge University Press
8. Nambiar, K.C. 2011. *Speaking Accurately. A Course in International Communication*. New Delhi : Foundation
9. Soundararaj, Francis. 2012. *Basics of Communication in English*. New Delhi: Macmillan
10. A textbook of English Phonetics for Indian Students by T. Balasubramanian (Macmillan)

DISTRIBUTION AND WEIGHTAGE OF MARKS

English Language Laboratory Practical Examination:

1. The practical examinations for the English Language Laboratory shall be conducted as per the University norms prescribed for the core engineering practical sessions.
2. For the Language lab sessions, there shall be a continuous evaluation during the year for 30 sessional marks and 70 semester-end Examination marks. Of the 30 marks, 20 marks shall be awarded for day-to-day work and 10 marks to be awarded by conducting Internal Lab Test(s). The year- end Examination shall be conducted by the teacher concerned with the help of another member of the staff of the same department of the same institution.

* * * * *