JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR

Course Structure for Computer Science and Engineering
B. Tech Course (2013-14)

IV - II Semester

S.No	Course code	Subject	Theory	Tu	Lab	Credits
1.	13A05801 13A05802	MOOC 1 1. Mobile Computing 2. Natural Language Processing	3	1	-	3
2.	13A05803 13A05804	MOOC 2 1. Parallel Algorithms 2. Real Time Systems	3	1	-	3
3.	13A05805 13A05806	MOOC 3 1. High Performance Computing 2. Programming for Everybody using Python	3	1	-	3
4.	13A05807	Technical Seminar	-	-	4	2
5.	13A05808	Project work	-	-	24	12
Total			**09**	**03**	**28**	**23**

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR

B.Tech. IV - II sem (C.S.E)

T	Tu	C
3	1	3

(13A05801) MOBILE COMPUTING
(MOOCS 1)

Course Objectives:

- Understand mobile ad hoc networks, design and implementation issues, and available solutions.
- Acquire knowledge of sensor networks and their characteristics.

Course Outcomes:

- Students able to use mobile computing more effectively
- Students gain understanding of the current topics in MANETs and WSNs, both from an industry and research point of views.
- Acquire skills to design and implement a basic mobile ad hoc or wireless sensor network via simulations.

UNIT-I:

Wireless LANS and PANS: Introduction, Fundamentals of WLANS, IEEE 802.11 Standards,
HIPERLAN Standard, Bluetooth, Home RF.

Wireless Internet:

Wireless Internet, Mobile IP, TCP in Wireless Domain, WAP, Optimizing Web over Wireless.

UNIT-II:

AD HOC Wireless Networks: Introduction, Issues in Ad Hoc Wireless Networks, AD Hoc Wireless Internet.

MAC Protocols for Ad Hoc Wireless Networks: Introduction, Issues in Designing a MAC protocol for Ad Hoc Wireless Networks, Design goals of a MAC Protocol for Ad Hoc Wireless Networks, Classifications of MAC Protocols, Contention - Based Protocols, Contention - Based Protocols with reservation Mechanisms, Contention – Based MAC Protocols with Scheduling Mechanisms, MAC Protocols that use Directional Antennas, Other MAC Protocols.

UNIT -III:

Routing Protocols: Introduction, Issues in Designing a Routing Protocol for Ad Hoc Wireless
Networks, Classification of Routing Protocols, Table –Driven Routing Protocols, On – Demand
Routing Protocols, Hybrid Routing Protocols, Routing Protocols with Efficient Flooding Mechanisms, Hierarchical Routing Protocols, Power – Aware Routing Protocols.

Transport Layer and Security Protocols: Introduction, Issues in Designing a Transport Layer
Protocol for Ad Hoc Wireless Networks, Design Goals of a Transport Layer Protocol for Ad Hoc
Wireless Networks, Classification of Transport Layer Solutions, TCP Over Ad Hoc Wireless Networks, Other Transport Layer Protocol for Ad Hoc Wireless Networks, Security in Ad Hoc Wireless Networks, Network Security Requirements, Issues and Challenges in Security Provisioning, Network Security Attacks, Key Management, Secure Routing in Ad Hoc Wireless Networks.

UNIT –IV:

Quality of Service: Introduction, Issues and Challenges in Providing QoS in Ad Hoc Wireless
Networks, Classification of QoS Solutions, MAC Layer Solutions, Network Layer Solutions, QoS Frameworks for Ad Hoc Wireless Networks.

Energy Management: Introduction, Need for Energy Management in Ad Hoc Wireless Networks, Classification of Ad Hoc Wireless Networks, Battery Management Schemes, Transmission Power Management Schemes, System Power Management Schemes.

UNIT –V:

Wireless Sensor Networks: Introduction, Sensor Network Architecture, Data Dissemination, Data Gathering, MAC Protocols for Sensor Networks, Location Discovery, Quality of a Sensor Network, Evolving Standards, Other Issues.

TEXT BOOKS:

1. Ad Hoc Wireless Networks: Architectures and Protocols - C. Siva Ram Murthy and B.S.Manoj, PHI, 2004.
2. Wireless Ad- hoc and Sensor Networks: Protocols, Performance and Control - Jagannathan Sarangapani, CRC Press

REFERENCE BOOKS:

1. Ad hoc Mobile Wireless Networks – Subir Kumar sarkar, T G Basvaraju, C Puttamadappa, Auerbach Publications,2012.
2. Wireless Sensor Networks - C. S. Raghavendra, Krishna M. Sivalingam, 2004, Springer.
3. Ad- Hoc Mobile Wireless Networks: Protocols & Systems, C.K. Toh , Pearson Education.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR

B.Tech. IV - II sem (C.S.E)

T	Tu	C
3	1	3

(13A05802) NATURAL LANGUAGE PROCESSING (MOOCS 1)

Objectives

- Understand and apply fundamental algorithms and techniques in the area of natural language processing (NLP).
- Understand approaches to syntax and semantics in NLP.
- Understand current methods for statistical approaches to machine translation.
- Understand language modeling.
- Understand machine learning techniques used in NLP.

Outcomes:

- Show sensitivity to linguistic phenomena and an ability to model them with formal grammars.
- Ability to design, implement and analyze NLP algorithms.

UNIT – I

Introduction to Natural Language Understanding, Syntactic Processing: Grammars and Parsing

UNIT-II:

Features and Augmented Grammars, Toward Efficient Parsing, Ambiguity Resolution

UNIT –III

Statistical Methods: Probabilistic Context-Free Grammars, Best-First Parsing.

UNIT-IV

Semantic Interpretation: Linking Syntax and Semantics, Ambiguity Resolution, other Strategies for Semantic Interpretation.

UNIT-V

Context and World Knowledge: Using World Knowledge, Discourse Structure, Defining a Conversational Agent.

TEXT BOOK:

1. Natural Language Understanding – James Allen, Second Edition, Pearson Education.

REFERENCE BOOKS:

1. Speech and Language Processing – Daniel Jurafsky, James H.Martin.
2. Foundations of Statistical Natural Language Processing – Christopher Manning, Hinrich Schutze, MIT Press.
3. Charniack, Eugene, Statistical Language Learning, MIT Press, 1993.
4. Jurafsky, Dan and Martin, James, Speech and Language Processing, 2nd Edition, Prentice Hall, 2013-2014
5. Manning, Christopher and Henrich, Schutze, Foundations of Statistical Natural Language Processing, MIT Press, 1999.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR

B.Tech. IV - II sem (C.S.E)

T	Tu	C
3	1	3

(13A05803) PARALLEL ALGORITHMS
(MOOCS 2)

Course Objective:
The objective of this course is to make the students

- Familiar with the efficient parallel algorithms related to many areas of computer science:
  expression computation, sorting, graph-theoretic problems, etc.
- Familiar with the basic issues of implementing parallel algorithms.
- Familiar with the fundamentals of discrete probability theory;
- able to know the basic randomized algorithms and to analyze selected randomized algorithms;
- Familiar with the theory of Markov chains and their algorithmic applications; knowledgeable about selected randomized data structures;

Course Outcomes:
Students who complete the course will have demonstrated the ability to do the following:

- Argue the correctness of algorithms using inductive proofs and invariants.
- Analyze worst-case running times of algorithms using asymptotic analysis.
- Explain the different ways to analyze parallel and randomized algorithms.
- Compare between different randomized data structures. Pick an appropriate data structure for a design situation.

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

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

UNIT III
Parallel Sorting Networks, Parallel Merging Algorithms on CREW/EREW/MCC, Parallel Sorting Networks on CREW/EREW/MCC/, linear array
Parallel Searching Algorithm, Kth element, Kth element in X+Y on PRAM, Parallel Matrix, Transportation and Multiplication Algorithm on PRAM, MCC, Vector-Matrix Multiplication, Solution of Linear Equation, Root finding.

UNIT IV
Randomized Algorithms: Example, Randomized Quicksort and Mincut Algorithms, Moments and Deviations Markov and Chebyshev Inequalities; Chernoff Bounds, martingales, Markov Chains and Random walks

UNIT V
Randomized Data Structures, Randomized Search Trees, Game tree; Hashing, Random Graphs, Random Walks in graphs, Derandomization

Text Books :
1. Designing Efficient Algorithms for Parallel Computer, M.J. Quinn, McGrawHill.

2. Probability and Computing: Randomized algorithms and Probabilistic Analysis, Michael Mitzenmacher and Eli Upfal. Cambridge University Press, 2005

Reference Books :

1. The Design and Analysis of Parallel Algorithms, S.G.Akl, PHI, 1989.
2. Randomized Algorithms, Rajeev Motwani and Prabhakar Raghavan, Cambridge University Press.
3. Design and Analysis of Randomized Algorithms: Introduction to Design Paradigms. Juraj Hromkovic, Springer, 2005.
4. Introducti on to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes, F.T.Lei ghton, MK Publishers, San Mateo California, 1992

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR

B.Tech. IV - II sem (C.S.E)

T	Tu	C
3	1	3

(13A05804) REAL TIME SYSTEMS
(MOOCS 2)

Objectives:

- Acquire skills necessary to design and develop embedded applications by means of real-time operating systems
- Understand embedded real-time operating systems

Course Outcomes:

- Characterize real-time systems and describe their functions
- Analyze, design and implement a real-time system
- Apply formal methods to the analysis and design of real-time systems
- Apply formal methods for scheduling real-time systems
- Characterize and describe reliability and fault tolerance issues and approaches.

Unit-1

Typical Real time Applications: Digital control, High-level control, Signal processing, other Real-time Applications.

Hard versus Soft Real-Time Systems: Jobs and processors, Release time, dead lines and Timing constraints, Hard and soft timing constraints, Hard Real time systems, Soft Real-time Systems.

A Reference Model of Real Time Systems: Processors and resources, Temporal parameters of Real time workload, periodic task model, precedence constraints and data dependency, Functional parameter, Resource Parameters of Jobs and Parameters of Resources, Scheduling Hierarchy.

Commonly used Approaches to real time Scheduling: Clock-Driven Approach, Weighted Round-Robin Approach, Priority driven Approach, Dynamic vs Static Systems, Effective release time and deadlines, Optimality of the EDF and LST algorithms, Nonoptimality of the EDF and LST algorithms, Challenges in validating timing constraints in priority driven System, Off line vs On line scheduling, summary.

Unit-2

Clock-Driven Scheduling: Notations and Assumptions, static, Timer-Driven scheduler, General Structure of the Cyclic Scheduler, Improving the average response time of Aperiodic Jobs, Scheduling sporadic Jobs, Practical considerations and generalizations, Algorithm for generating Static Schedules, Pros and cons of Clock-driven scheduling, summary.

Unit-3

Priority-Driven Scheduling of periodic Tasks : Static Assumption, Fixed-priority vs Dynamic-priority Algorithms, Maximum Schedulable Utilization, Optimality of the RM and

DM Algorithms, A Schedulability test for Fixed-priority tasks with Short Response time, A Schedulability test for Fixed-priority tasks with arbitrary Response time, Sufficient Schedulability conditions for the RM and DM Algorithms, summary.

Unit-4

Scheduling Aperiodic and Sporadic Jobs in Priority Driven Systems: Assumptions and approaches, Diferrable servers, Sporadic Servers, Constant utilization, total bandwidth and weighted fair –Queueing servers, Slack stealing in Dead-line Driven System, Stack stealing in Fixed-priority systems, Scheduling of sporadic jobs, Real-time performance for jobs with soft timing constraints, A two-level scheme for Integrated scheduling.

Unit-5

Resources and Resource access control: Assumptions on Resources and their usage, Effects of Resource contention and resource access control, NonPreemptive critical section, Basic Priority inheritance protocol, Basic Priority ceiling protocol, Stack –based, Priority ceiling protocol, Use of priority ceiling protocol in Dynamic priority systems, pre-emption ceiling protocol, Controlling accesses to Multiple unit Resources, Controlling concurrent accesses to data objects.

Multiprocessor Scheduling, Resource access control, and Synchronization: Model of Multiprocessor and Distributed Systems, Task assignment, Multiprocessor Priority ceiling protocol, Elements of Scheduling Algorithms for End-to-End Periodic Tasks, Schedulability of Fixed-priority End-to-End periodic Tasks, End to End tasks in heterogeneous Systems, Predictability and validation of Dynamic Multiprocessor Systems, Summary

Text Book:

1) “Real-Time Systems” by Jane W.S Liu, Pearson Edition, 2006.

Reference Text Book:

1. Real-Time Systems: Scheduling, Analysis, and Verification, Cheng, A. M. K.: Wiley, 2002.
2. Z.: Scheduling in Real-Time Systems, by Cottet, F., Delacroix, J., Kaiser, C., Mammeri John Wiley & Sons, 2002.
3. Real-Time Systems, C. M., Shin, K. G. McGraw-Hill, Krishna 1997.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR

B.Tech. IV - II sem (C.S.E)

T	Tu	C
3	1	3

(13A05805) HIGH PERFORMANCE COMPUTING
(MOOCS 3)

Objectives:

This course deals with two interrelated issues in high-performance computing:

- Fundamental concepts and techniques in parallel computation structuring and design, including parallelization methodologies and paradigms, parallel programming models, their implementation, and related cost models;
- Architectures of high-performance computing systems, including shared memory multiprocessors, distributed memory multi computers, clusters, and others.

Course Outcomes:

At the end of the module, a student will have an understanding of

- Ability to measure, analyze, and assess the performance of HPC applications and their supporting hardware.
- Ability to design the parallel algorithms.

UNIT I:

Introduction to Parallel Computing:
Motivating parallelism, Scope of Parallel Computing.
Parallel Programming Platforms:
Implicit Parallelism: Trends in Microprocessor Architectures, Limitations of Memory System Performance, Dichotomy of Parallel Computing Platforms, Physical Organization of Parallel Platforms, Communication costs in Parallel Machines, Routing Mechanisms for Interconnection Networks, Impact of Process –Processor Mapping and Mapping Techniques.

UNIT II:
Principles of Parallel Algorithm Resign:
Preliminaries, Decomposition Techniques, Characteristics of tasks and Interactions, Mapping Techniques for Load Balancing, Methods for containing Interaction Overheads, Parallel Algorithm Models.
Basic Communication Operators:
One-to-All Broadcast and All-to-One Reduction, All-to-All Broadcast and Reduction, All-Reduce and Prefix-Sum Operations, Scatter and Gather, All-to-All Personalized Communication, Circular Shift, Improving the speed of Some Communication Operations.

UNIT III:
Programming Using Message-Passing Paradigm:
Principles of Message-Passing, The Building Blocks: Send and Receive Operations, MPI: The Message Passing Interface, Topologies and Embedding, Overlapping Communication

with computation, Collective Communication and Computation Operations, Groups and Communications.

Programming Shared Address Space Platforms:

Thread Basics, Why Threads?, The POSIX Thread, Thread Basics: Creation and Termination, Synchronization Primitives in Pthreads, Collecting Thread and Synchronization Attributes, Thread Cancellation, Tips for designing Asynchronous Programs, OpenMP: a Standard for Directive based parallel programming.

UNIT IV:

Dense Matrix Algorithms:

Matrix-Vector Multiplication, Matrix-Matrix Multiplication, Solving a System for Linear Equations.

Sorting:

Issues in Sorting on Parallel Computers, Sorting Networks, Bubble Sort and its Variants, Quick Sort, Bucket and Sample Sort, Other Sorting Algorithms.

UNIT V:

Graph Algorithms:

Definitions and Representation, Minimum Spanning Tree: Prim's Algorithm, Single Source Shortest path: Dijkstra's Algorithm, All-Pair Shortest Paths, Transitive Closure, Connected components, Algorithms for Space Graphs.

Search Algorithms For Discrete Optimization Problems:

Definitions and Examples, Sequential Search Algorithms, Search Overhead Factor, Parallel Depth-First Search, Parallel Best-First Search, Speedup Anomalies in Parallel Search Algorithms.

TEXT BOOKS:

1."Introduction to Parallel Computing" by Ananth Grama, Anshul Guptha, George Karypis and Vipin Kumar. Pearson, 2nd Edition.

REFFRENCE BOOKS:

1. "Parallel Programming- Techniques and applications using networked workstations and parallel computers" by Barry Wilkinson, Michael Allen , Pearson Education, 2nd Edition 2007.

2."Multi Core Programming – Increasing Performance through Software Multi-threading" by Shameem Akhter and Jason Roberts, Intel Press 2006.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR

B.Tech. IV - II sem (C.S.E)

T	Tu	C
3	1	3

(13A05806) PYTHON PROGRAMMING
(MOOCS 3)

OBJECTIVES:

- Introduction to Scripting Language
- Exposure to various problems solving approaches of computer science

UNIT – I:
Introduction: History of Python, Need of Python Programming, Applications Basics of Python Programming Using the REPL(Shell), Running Python Scripts, Variables, Assignment, Keywords, Input-Output, Indentation.

UNIT – II:
Types, Operators and Expressions: Types - Integers, Strings, Booleans; Operators- Arithmetic Operators, Comparison (Relational) Operators, Assignment Operators, Logical Operators, Bitwise Operators, Membership Operators, Identity Operators, Expressions and order of evaluations Control Flow- if, if-elif-else, for, while, break, continue, pass

UNIT – III:
Data Structures Lists - Operations, Slicing, Methods; Tuples, Sets, Dictionaries, Sequences. Comprehensions.

UNIT – IV:
Functions - Defining Functions, Calling Functions, Passing Arguments, Keyword Arguments, Default Arguments, Variable-length arguments, Anonymous Functions, Fruitful Functions(Function Returning Values), Scope of the Variables in a Function - Global and Local Variables.

Modules: Creating modules, import statement, from ..import statement, name spacing,

Python packages, Introduction to PIP, Installing Packages via PIP, Using Python Packages Error and Exceptions: Difference between an error and Exception, Handling Exception, try except block, Raising Exceptions, User Defined Exceptions

UNIT – V:
Object Oriented Programming OOP in Python: Classes, 'self variable', Methods, Constructor Method, Inheritance, Overriding Methods, Datahiding,

UNIT – VI:

Brief Tour of the Standard Library - Operating System Interface - String Pattern Matching, Mathematics, Internet Access, Dates and Times, Data Compression, Multithreading, GUI Programming, Turtle Graphics

Testing: Why testing is required ?, Basic concepts of testing, Unit testing in Python, Writing Test cases, Running Tests.

OUTCOMES:

- Making Software easily right out of the box.
- Experience with an interpreted Language.
- To build software for real needs.
- Prior Introduction to testing software

TEXT BOOKS

1. Python Programming: A Modern Approach, Vamsi Kurama, Pearson
2. Learning Python, Mark Lutz, Orielly

Reference Books:

1. Think Python, Allen Downey, Green Tea Press
2. Core Python Programming, W.Chun, Pearson.
3. Introduction to Python, Kenneth A. Lambert, Cengage