VEL TECH MULTITECH
Dr RANGARAJAN Dr.SAKUNTHALA ENGINEERING COLLEGE

(An ISO 9001: 2008 Certified Institution)
(Owned by 'VEL Shree R. Rangarajan
Dr. Sakunthala Rangarajan Educational Academy)
(Approved by AICTE, New Delhi &
Govt. of Tamil Nadu and affiliated to Anna University)

SYLLABUS

WEEKLY SCHEDULE

IV SEMESTER 2015-16

DEPARTMENT OF IT

IV YEAR DEGREE COURSE

#60, Avadi – Alamathi Road,
Chennai – 600062
Telefax – 044-26841061
E-mail: veltech@md3.vsnl.net.in
Website: www.veltechmultitech.org

SEM: IV YEAR: II

ACADEMIC YEAR (2015-16)

Regulation 2013

S. No	WEEKS	DATE	
		FROM	TO
1	WEEK1	18.01.16	23.01.16
2	WEEK2	25.01.16	30.01.16
3	WEEK3	01.02.16	06.02.16
4	WEEK4	08.02.16	13.02.16
5	WEEK5	15.02.16	20.02.16
6	WEEK6	22.02.16	27.02.16
7	WEEK7	29.02.16	05.03.16
8	WEEK8	07.03.16	12.03.16
9	WEEK9	16.03.16	20.03.16
10	WEEK10	22.03.16	26.03.16
11	WEEK11	27.03.16	02.04.16
12	WEEK12	03.04.16	09.04.16
13	WEEK13	11.04.16	16.04.16
14	WEEK14	18.04.16	23.04.16
15	WEEK 15	25.04.16	30.04.16

CONTENTS

TEST SCHEDULE

S.No	SUB CODE	SUBJECT NAME	UNIT TEST I	UNIT TEST II	PRE MODEL EXAM	MODEL EXAM
1	MA6453	Probability and Queuing Theory	01.02.16	15.02.16	29.02.06	01.04.16
2	EC6504	Microprocessor and Microcontroller	02.02.16	16.02.16	01.03.16	04.04.16
3	CS6402	Design and Analysis of Algorithms	03.02.16	17.02.16	02.03.16	06.04.16
4	CS6401	Operating Systems	04.02.16	18.02.16	03.03.16	08.04.16
5	CS6403	Software Engineering	05.02.16	19.02.16	04.03.16	11.04.16

MA6453 PROBABILITY AND QUEUING THEORY

OBJECTIVES:

- To provide the required mathematical support in real life problems and develop probabilistic models which can be used in several areas of science and engineering.

WEEK-1

UNIT I RANDOM VARIABLES

Discrete and continuous random variables – Moments – Moment generating functions – Binomial, Poisson, Geometric, Uniform, Exponential, Gamma and Normal distributions.

WEEK-2 Unit Test-1

UNIT II TWO - DIMENSIONAL RANDOM VARIABLES

Joint distributions

WEEK-3

Marginal and conditional distributions.

WEEK-4

Covariance – Correlation and Linear regression **.**

WEEK-5

Transformation of random variables.

WEEK-6 Unit Test-2

UNIT III RANDOM PROCESSES

Classification – Stationary process

WEEK-7

Markov process - Poisson process – Discrete parameter Markov chain

WEEK-8

Chapman Kolmogorov equations – Limiting distributions.

WEEK-9 Unit Test-3

UNIT IV QUEUEING MODELS

Markovian queues – Birth and Death processes.

WEEK-10
Single and multiple server queueing models – Little's formula - Queues with finite waiting rooms.

WEEK-11
Queues with impatient customers: Balking and reneging.

WEEK-12 Unit Test-4

UNIT V ADVANCED QUEUEING MODELS
Finite source models - M/G/1 queue.

WEEK-13
Pollaczek Khinchin formula - M/D/1 and M/EK/1 as special cases.

WEEK-14
Series queues – Open Jackson networks.

WEEK-15 Unit Test-5

WEEK-16 & 17 Model Exam

OUTCOMES:
- The students will have a fundamental knowledge of the probability concepts.
- Acquire skills in analyzing queueing models.
- It also helps to understand and characterize phenomenon which evolve with respect to time in a probabilistic manner.

TEXT BOOKS:

1. Ibe. O.C., "Fundamentals of Applied Probability and Random Processes", Elsevier, 1st Indian Reprint, 2007.
2. Gross. D. and Harris. C.M., "Fundamentals of Queueing Theory", Wiley Student edition, 2004.

REFERENCES:

1. Robertazzi, "Computer Networks and Systems: Queueing Theory and Performance Evaluation", 3rd Edition, Springer, 2006.
2. Taha. H.A., "Operations Research", 8th Edition, Pearson Education, Asia, 2007.

3. Trivedi.K.S., "Probability and Statistics with Reliability, Queueing and Computer Science Applications", 2nd Edition, John Wiley and Sons, 2002.
4. Hwei Hsu, "Schaum's Outline of Theory and Problems of Probability, Random Variables and Random Processes", Tata McGraw Hill Edition, New Delhi, 2004.
5. Yates. R.D. and Goodman. D. J., "Probability and Stochastic Processes", 2nd Edition, Wiley India Pvt. Ltd., Bangalore, 2012.

EC6504 MICROPROCESSOR AND MICROCONTROLLER

OBJECTIVES:

The student should be made to:

- Study the Architecture of 8086 microprocessor.
- Learn the design aspects of I/O and Memory Interfacing circuits.
- Study about communication and bus interfacing.
- Study the Architecture of 8051 microcontroller.

WEEK-1

UNIT I THE 8086 MICROPROCESSOR

Introduction to 8086 – Microprocessor architecture – Addressing modes - Instruction set and assembler directives – Assembly language programming – Modular Programming - Linking and Relocation - Stacks - Procedures – Macros – Interrupts and interrupt service routines – Byte and String Manipulation.

WEEK-2 Unit Test-1

UNIT II 8086 SYSTEM BUS STRUCTURE

8086 signals – Basic configurations – System bus timing –System design using 8086.

WEEK-3

IO programming – Introduction to Multiprogramming – System Bus Structure – Multiprocessor configurations.

WEEK-4

Coprocessor, Closely coupled and loosely Coupled configurations.

WEEK-5

Introduction to advanced processors.

WEEK-6 Unit Test-2

UNIT III I/O INTERFACING

Memory Interfacing and I/O interfacing - Parallel communication interface – Serial communication interface.

WEEK-7

D/A and A/D Interface - Timer – Keyboard /display controller – Interrupt controller, DMA controller.

WEEK-8

Programming and applications Case studies: Traffic Light control, LED display, LCD display, Keyboard display interface and Alarm Controller.

WEEK-9 Unit Test-3

UNIT IV MICROCONTROLLER

Architecture of 8051 – Special Function Registers(SFRs)

WEEK-10

I/O Pins Ports and Circuits – Instruction set

WEEK-11

Addressing modes - Assembly language programming.

WEEK-12 Unit Test-4

UNIT V INTERFACING MICROCONTROLLER

Programming 8051 Timers - Serial Port Programming

WEEK-13

Interrupts Programming – LCD & Keyboard Interfacing - ADC, DAC & Sensor Interfacing

WEEK-14

External Memory Interface- Stepper Motor and Waveform generation.

WEEK-15 Unit Test-5

WEEK-16 & 17 Model Exam

OUTCOMES:

At the end of the course, the student should be able to:

- Design and implement programs on 8086 microprocessor.
- Design I/O circuits.
- Design Memory Interfacing circuits.
- Design and implement 8051 microcontroller based systems.

TEXT BOOKS:

1. Yu-Cheng Liu, Glenn A.Gibson, "Microcomputer Systems: The 8086 / 8088 Family - Architecture, Programming and Design", Second Edition, Prentice Hall of India, 2007.
2. Mohamed Ali Mazidi, Janice Gillispie Mazidi, Rolin McKinlay, "The 8051 Microcontroller and Embedded Systems: Using Assembly and C", Second Edition, Pearson Education, 2011.

REFERENCE:

1. Doughlas V.Hall, "Microprocessors and Interfacing, Programming and Hardware", TMH,2012

CS6402 DESIGN AND ANALYSIS OF ALGORITHMS

OBJECTIVES:

The student should be made to:

- Learn the algorithm analysis techniques.
- Become familiar with the different algorithm design techniques.
- Understand the limitations of Algorithm power.

WEEK-1

UNIT I INTRODUCTION

Notion of an Algorithm – Fundamentals of Algorithmic Problem Solving – Important Problem Types – Fundamentals of the Analysis of Algorithm Efficiency – Analysis Framework – Asymptotic Notations and its properties – Mathematical analysis for Recursive and Non-recursive algorithms.

WEEK-2 Unit Test-1

UNIT II BRUTE FORCE AND DIVIDE-AND-CONQUER

Brute Force - Closest-Pair and Convex-Hull Problems

WEEK-3
Exhaustive Search - Traveling Salesman Problem - Knapsack Problem - Assignment problem.

WEEK-4
Divide and conquer methodology – Merge sort – Quick sort – Binary search – Multiplication of Large Integers

WEEK-5
Strassen's Matrix Multiplication-Closest-Pair and Convex-Hull Problems.

WEEK-6 Unit Test-2

UNIT III DYNAMIC PROGRAMMING AND GREEDY TECHNIQUE
Computing a Binomial Coefficient – Warshall's and Floyd' algorithm – Optimal Binary Search Trees.

WEEK-7
Knapsack Problem and Memory functions. Greedy Technique–

WEEK-8
Prim's algorithm- Kruskal's Algorithm- Dijkstra's Algorithm-Huffman Trees.

WEEK-9 Unit Test-3

UNIT IV ITERATIVE IMPROVEMENT
The Simplex MeFthod-The Maximum-Flow Problem

WEEK-10
Maximm Matching in Bipartite Graphs

WEEK-11
The Stable marriage Problem.

WEEK-12 Unit Test-4

UNIT V COPING WITH THE LIMITATIONS OF ALGORITHM POWER
Limitations of Algorithm Power-Lower-Bound Arguments-Decision Trees-P, NP and NP-Complete Problems

WEEK-13
Coping with the Limitations - Backtracking – n-Queens problem – Hamiltonian Circuit Problem – Subset Sum Problem-Branch and Bound – Assignment problem – Knapsack Problem

WEEK-14
Traveling Salesman Problem- Approximation Algorithms for NP – Hard Problems – Traveling Salesman problem – Knapsack problem.

WEEK-15 Unit Test-5

WEEK-16 & 17 Model Exam

OUTCOMES:
At the end of the course, the student should be able to:

- Design algorithms for various computing problems.
- Analyze the time and space complexity of algorithms.
- Critically analyze the different algorithm design techniques for a given problem.
- Modify existing algorithms to improve efficiency.

TEXT BOOK:

1. Anany Levitin, "Introduction to the Design and Analysis of Algorithms", Third Edition, Pearson Education, 2012.

REFERENCES:

2. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", Third Edition, PHI Learning Private Limited, 2012.
3. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, "Data Structures and Algorithms", Pearson Education, Reprint 2006.
4. Donald E. Knuth, "The Art of Computer Programming", Volumes 1& 3 Pearson Education, 2009.
5. Steven S. Skiena, "The Algorithm Design Manual", Second Edition, Springer, 2008.
6. http://nptel.ac.in/

CS6401 OPERATING SYSTEMS

OBJECTIVES:

The student should be made to:

- Study the basic concepts and functions of operating systems.
- Understand the structure and functions of OS.
- Learn about Processes, Threads and Scheduling algorithms.
- Understand the principles of concurrency and Deadlocks.
- Learn various memory management schemes.
- Study I/O management and File systems.
- Learn the basics of Linux system and perform administrative tasks on Linux Servers.

WEEK-1

UNIT I OPERATING SYSTEMS OVERVIEW

Computer System Overview-Basic Elements, Instruction Execution, Interrupts, Memory Hierarchy, Cache Memory, Direct Memory Access, Multiprocessor and Multicore Organization. Operating system overview-objectives and functions, Evolution of Operating System - Computer System Organization- Operating System Structure and Operations- System Calls, System Programs, OS Generation and System Boot.

WEEK-2 Unit Test-1

UNIT II PROCESS MANAGEMENT

Processes-Process Concept, Process Scheduling, Operations on Processes,

WEEK-3

Interprocess Communication; Threads- Overview, Multicore Programming, Multithreading Models; Windows 7 - Thread and SMP Management.

WEEK-4

Process Synchronization - Critical Section Problem, Mutex Locks

WEEK-5

Semophores, Monitors; CPU Scheduling and Deadlocks.

WEEK-6 Unit Test-2

UNIT III STORAGE MANAGEMENT

Main Memory-Contiguous Memory Allocation, Segmentation, Paging

WEEK-7
32 and 64 bit architecture Examples; Virtual Memory- Demand Paging, Page Replacement, Allocation.

WEEK-8
Thrashing; Allocating Kernel Memory, OS Examples.

WEEK-9 Unit Test-3

UNIT IV I/O SYSTEMS
Mass Storage Structure- Overview, Disk Scheduling and Management; File System Storage

WEEK-10
File Concepts, Directory and Disk Structure, Sharing and Protection;
File System Implementation, File System Structure

WEEK-11
Directory Structure, Allocation Methods, Free Space Management; I/O Systems.

WEEK-12 Unit Test-4

UNIT V CASE STUDY
Linux System- Basic Concepts; System Administration-Requirements for Linux System Administrator,

WEEK-13
Setting up a LINUX Multifunction Server, Domain Name System, Setting Up Local Network Services;

WEEK-14
Virtualization- Basic Concepts, Setting Up Xen, VMware on Linux Host and Adding Guest OS.

WEEK-15 Unit Test-5

WEEK-16 & 17 Model Exam

OUTCOMES:

At the end of the course, the student should be able to:

- Design various Scheduling algorithms.
- Apply the principles of concurrency.
- Design deadlock, prevention and avoidance algorithms.
- Compare and contrast various memory management schemes.
- Design and Implement a prototype file systems.
- Perform administrative tasks on Linux Servers.

TEXT BOOK:

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, "Operating System Concepts", 9th Edition, John Wiley and Sons Inc., 2012.

REFERENCES:

2. William Stallings, "Operating Systems – Internals and Design Principles", 7th Edition, Prentice Hall, 2011.
3. Andrew S. Tanenbaum, "Modern Operating Systems", Second Edition, Addison Wesley, 2001.
4. Charles Crowley, "Operating Systems: A Design-Oriented Approach", Tata McGraw Hill Education", 1996.
5. D M Dhamdhere, "Operating Systems: A Concept-Based Approach", Second Edition, Tata McGraw-Hill Education, 2007.
6. http://nptel.ac.in/.

CS6403 SOFTWARE ENGINEERING

OBJECTIVES:

The student should be made to:

- Understand the phases in a software project
- Understand fundamental concepts of requirements engineering and Analysis Modeling.
- Understand the major considerations for enterprise integration and deployment.
- Learn various testing and maintenance measures

WEEK-1

UNIT I SOFTWARE PROCESS AND PROJECT MANAGEMENT

Introduction to Software Engineering, Software Process, Perspective and Specialized Process Models – Software Project Management: Estimation – LOC and FP Based Estimation, COCOMO Model – Project Scheduling – Scheduling, Earned Value Analysis - Risk Management.

WEEK-2 Unit Test-1

UNIT II REQUIREMENTS ANALYSIS AND SPECIFICATION

Software Requirements: Functional and Non-Functional, User requirements, System requirements

WEEK-3

Software Requirements Document – Requirement Engineering Process: Feasibility Studies

WEEK-4

Requirements elicitation and analysis, requirements validation, requirements management

WEEK-5

Classical analysis: Structured system Analysis, Petri Nets- Data Dictionary.

WEEK-6 Unit Test-2

UNIT III SOFTWARE DESIGN

Design process – Design Concepts-Design Model– Design Heuristic – Architectural Design.

WEEK-7

Architectural styles, Architectural Design, Architectural Mapping using Data Flow- User Interface.

WEEK-8

Design: Interface analysis, Interface Design –Component level Design: Designing Class based components, traditional Components.

WEEK-9 Unit Test-3

UNIT IV TESTING AND IMPLEMENTATION

Software testing fundamentals-Internal and external views of Testing

WEEK-10

White box testing - basis path testing-control structure testing-black box testing-Regression Testing – Unit Testing – Integration Testing

WEEK-11
Validation Testing – System Testing And Debugging – Software Implementation Techniques**:** Coding practices-Refactoring.

WEEK-12 Unit Test-4

UNIT V PROJECT MANAGEMENT
Estimation – FP Based, LOC Based, Make/Buy Decision, COCOMO II –

WEEK-13
Planning – Project Plan,
Planning Process, RFP Risk Management – Identification, Projection, RMMM

WEEK-14
Scheduling and Tracking –Relationship between people and effort, Task Set & Network, Scheduling, EVA – Process and Project Metrics.

WEEK-15 Unit Test-5

WEEK-16 & 17 Model Exam

OUTCOMES:
At the end of the course, the student should be able to

- Identify the key activities in managing a software project.
- Compare different process models.
- Concepts of requirements engineering and Analysis Modeling.
- Apply systematic procedure for software design and deployment.
- Compare and contrast the various testing and maintenance

TEXT BOOKS:

1. Roger S. Pressman, "Software Engineering – A Practitioner's Approach", Seventh Edition, Mc Graw-Hill International Edition, 2010.

REFERENCES:

2. Ian Sommerville, "Software Engineering", 9th Edition, Pearson Education Asia, 2011.
3. Rajib Mall, "Fundamentals of Software Engineering", Third Edition, PHI Learning Private Limited, 2009.
4. Pankaj Jalote, "Software Engineering, A Precise Approach", Wiley India, 2010.

5. Kelkar S.A., “Software Engineering”, Prentice Hall of India Pvt Ltd, 2007.
6. Stephen R.Schach, “Software Engineering”, Tata McGraw-Hill Publishing Company Limited, 2007.
7. http://nptel.ac.in/.

IT6411 MICROPROCESSOR AND MICROCONTROLLER LABORATORY

OBJECTIVES:

The student should be made to:

- Introduce ALP concepts and features
- Write ALP for arithmetic and logical operations in 8086 and 8051
- Differentiate Serial and Parallel Interface
- Interface different I/Os with Microprocessors
- Be familiar with MASM

LIST OF EXPERIMENTS:

8086 Programs using kits and MASM

1. Basic arithmetic and Logical operations
2. Move a data block without overlap
3. Code conversion, decimal arithmetic and Matrix operations.
4. Floating point operations, string manipulations, sorting and searching
5. Password checking, Print RAM size and system date
6. Counters and Time Delay

Peripherals and Interfacing Experiments

7. Traffic light control
8. Stepper motor control
9. Digital clock
10. Key board and Display
11. Printer status
12. Serial interface and Parallel interface
13. A/D and D/A interface and Waveform Generation

8051 Experiments using kits and MASM

14. Basic arithmetic and Logical operations
15. Square and Cube program, Find 2's complement of a number
16. Unpacked BCD to ASCII

OUTCOMES:

At the end of the course, the student should be able to:

- Write ALP Programmes for fixed and Floating Point and Arithmetic
- Interface different I/Os with processor
- Generate waveforms using Microprocessors
- Execute Programs in 8051
- Explain the difference between simulator and Emulator

IT6412 OPERATING SYSTEMS LABORATORY

OBJECTIVES:

The student should be made to:

- Learn shell programming and the use of filters in the UNIX environment.
- Be exposed to programming in C using system calls.
- Learn to use the file system related system calls.
- Be exposed to process creation and inter process communication.
- Be familiar with implementation of CPU Scheduling Algorithms, page replacement algorithms and Deadlock avoidance

LIST OF EXPERIMENTS:

1. Basics of UNIX commands.
2. Shell Programming.
3. Implement the following CPU scheduling algorithms

a) Round Robin b) SJF c) FCFS d) Priority

4. Implement all file allocation strategies

a) Sequential b) Indexed c) Linked

5. Implement Semaphores
6. Implement all File Organization Techniques

a) Single level directory b) Two level c) Hierarchical d) DAG

7. Implement Bankers Algorithm for Dead Lock Avoidance
8. Implement an Algorithm for Dead Lock Detection
9. Implement e all page replacement algorithms

a) FIFO b) LRU c) LFU

10. Implement Shared memory and IPC
11. Implement Paging Technique of memory management.
12. Implement Threading & Synchronization Applications

REFERENCE:

1. spoken-tutorial.org

OUTCOMES:

At the end of the course, the student should be able to

- Implement deadlock avoidance, and Detection Algorithms
- Compare the performance of various CPU Scheduling Algorithm
- Critically analyze the performance of the various page replacement algorithms
- Create processes and implement IPC

IT6413 SOFTWARE ENGINEERING LABORATORY

OBJECTIVES:

- To understand the software engineering methodologies for project development.
- To gain knowledge about open source tools for Computer Aided Software Engineering.
- To develop an efficient software using case tools.

SOFTWARE REQUIRED:

Open source Tools: StarUML / UMLGraph / Topcased

Prepare the following documents for each experiment and develop the software using software engineering methodology.

1. Problem Analysis and Project Planning -Thorough study of the problem – Identify Project scope, Objectives and Infrastructure.

2. Software Requirement Analysis - Describe the individual Phases/modules of the project and
Identify deliverables.

3. Data Modelling - Use work products – data dictionary, use case diagrams and activity diagrams, build and test class diagrams, sequence diagrams and add interface to class diagrams.

4. Software Development and Debugging – implement the design by coding

5. Software Testing - Prepare test plan, perform validation testing, coverage analysis, memory leaks, develop test case hierarchy, Site check and site monitor.

Sample Experiments:

Academic domain

1. Course Registration System
2. Student marks analysing system

Railway domain

3. Online ticket reservation system
4. Platform assignment system for the trains in a railway station

Medicine domain

5. Expert system to prescribe the medicines for the given symptoms
6. Remote computer monitoring

Finance domain

7. ATM system
8. Stock maintenance

Human Resource management

9. Quiz System
10. E-mail Client system.

OUTCOMES:

Upon Completion of the course, the students should be able to:

- Use open source case tools to develop software.
- Analyze and design software requirements in efficient manner.