



VEL TECH MULTI TECH Dr RANGARAJAN Dr.SAKUNTHALA ENGINEERING COLLEGE

(An ISO 9001: 2008 Certified Institution)

(Owned by Vel Trust)

(Approved by Govt. of Tamil Nadu and affiliated to Anna University and
Accredited by NBA, New Delhi)



SYLLABUS

WEEKLY SCHEDULE

V SEMESTER

2014 - 2015

DEPARTMENT OF CSE

IV DEGREE COURSE

**42, Avadi – Alamathi Road,
Chennai – 600062**

Telefax – 044-26841061

E-mail: emailto@veltechmultitech.org

Website : www.veltechmultitech.org



WEEK DETAILS

SL.NO.	WEEK	FROM	TO
1	WEEK1	24-06-2014	27-06-2014
2	WEEK2	30-06-2014	04-07-2014
3	WEEK3	07-07-2014	11-07-2014
4	WEEK4	14-07-2014	18-07-2014
5	WEEK5	21-07-2014	25-07-2014
6	WEEK6	28-07-2014	01-08-2014
7	WEEK7	04-08-2014	08-08-2014
8	WEEK8	11-08-2014	14-08-2014
9	WEEK9	18-08-2014	22-08-2014
10	WEEK10	25-08-2014	28-08-2014
11	WEEK11	01-09-2014	05-09-2014
12	WEEK12	08-09-2014	12-09-2014
13	WEEK13	15-09-2014	19-09-2014
14	WEEK14	22-09-2014	26-09-2014
15	WEEK15	29-09-2014	01-10-2014
16	WEEK16	06-10-2014	10-10-2014
17	WEEK17	13-10-2014	17-10-2014
18	WEEK18	20-10-2014	24-10-2014
19	WEEK19	27-10-2014	31-10-2014

SUBJECT CONTENTS

SL.NO	SUBJECT CODE	SUBJECT NAME
THEORY		
1	CS2301	Software Engineering
2	MA 2265	Discrete Mathematics
3	CS2302	Computer Networks
4	CS2303	Theory of Computation
5	CS2304	System Software
6	CS2305	Programming Paradigms
PRACTICAL		
7	CS2307	Network Lab
8	CS2308	System software lab
9	CS2309	Java Lab

TEST / EXAM SCHEDULE

SL.NO	SUBJECT CODE	SUBJECT NAME	UNIT TEST I	UNIT TEST II	UNIT TEST III	UNIT TEST IV	UNIT TEST V
1	CS2301	Software Engineering	08/07/14 FN	30/07/14 FN	20/08/14 FN	09/09/14 FN	29/09/14 FN
2	MA 2265	Discrete Mathematics	08/07/14 AN	30/07/14 AN	20/08/14 AN	09/09/14 AN	29/09/14 AN
3	CS2302	Computer Networks	09/07/14 FN	31/07/14 FN	21/08/14 FN	10/09/14 FN	30/09/14 FN
4	CS2303	Theory of Computation	09/07/14 AN	31/07/14 AN	21/08/14 AN	10/09/14 AN	30/09/14 AN
5	CS2304	System Software	10/07/14 FN	01/08/14 FN	22/08/14 FN	11/09/14 FN	01/10/14 FN
6	CS2305	Programming Paradigms	10/07/14 AN	01/08/14 AN	22/08/14 AN	11/09/14 AN	01/10/14 AN

SL.NO	SUBJECT CODE	SUBJECT NAME	MODEL EXAM
1	CS2301	Software Engineering	13-10-2014
2	MA 2265	Discrete Mathematics	14-10-2014
3	CS2302	Computer Networks	15-10-2014
4	CS2303	Theory of Computation	16-10-2014
5	CS2304	System Software	17-10-2014
6	CS2305	Programming Paradigms	20-10-2014

CS2301: SOFTWARE ENGINEERING

UNIT I – SOFTWARE PRODUCT AND PROCESS

WEEK 1: Introduction – S/W Engineering Paradigm – Verification – Validation

WEEK 2: Life Cycle Models – System Engineering – Computer Based System

WEEK 3: Business Process Engineering Overview – Product Engineering Overview

UNIT II – SOFTWARE REQUIREMENTS

WEEK 4: UNIT TEST-I

Functional and Non-Functional – Software Document – Requirement Engineering Process

WEEK 5: Feasibility Studies – Software Prototyping Prototyping in the Software Process

WEEK 6: Data – Functional and Behavioral Models- Structured Analysis and Data Dictionary.

WEEK 7: UNIT TEST-II

UNIT III - ANALYSIS, DESIGN CONCEPTS AND PRINCIPLES

WEEK 8: Systems Engineering - Analysis Concepts - Design Process And Concepts:- Modular Design- Design Heuristic – Architectural Design

WEEK 9: Data Design – User Interface Design Real Time Software Design – System Design – Real Time Executives – Data Acquisition System – Monitoring And Control System.

UNIT IV – TESTING

WEEK 10:– UNIT TEST-III

Taxonomy Of Software Testing – Types Of S/W Test – Black Box Testing Testing Boundary Conditions – Structural Testing – Test Coverage Criteria Based On Data Flow Mechanisms

WEEK 11: Regression Testing – Unit Testing – Integration Testing – Validation Testing

WEEK12: System Testing And Debugging – Software Implementation Techniques

WEEK 13 : UNIT TEST-IV

**WEEK 14 :- UNIT V -SOFTWARE PROJECT MANAGEMENT
Measures And Measurements – ZIPF’s Law – Software Cost Estimation**

WEEK 15:– Function Point Models – COCOMO Model- Delphi Method – Scheduling – Earned Value Analysis – Error Tracking

WEEK 16: Software Configuration Management – Program Evolution Dynamics - Software Maintenance – Project Planning – Project Scheduling– Risk Management – CASE Tools

WEEK 17: UNIT TEST-V

WEEK 18: MODEL PRACTICAL EXAM

WEEK 19: MODEL EXAM

TEXT BOOK

- 1. Ian Sommerville, “Software engineering”, Seventh Edition, Pearson Education Asia, 2007.**
- 2. Roger S. Pressman, “Software Engineering – A practitioner’s Approach”, Sixth Edition, McGraw-Hill International Edition, 2005.**

REFERENCES

- 1. Watts S.Humphrey,"A Discipline for Software Engineering", Pearson Education, 2007.**
- 2. James F.Peters and Witold Pedrycz,"Software Engineering, An Engineering Approach", Wiley-India, 2007.**
- 3. Stephen R.Schach, " Software Engineering", Tata McGraw-Hill Publishing Company Limited, 2007.**
- 4. S.A.Kelkar,"Software Engineering", Prentice Hall of India Pvt, 2007.**

MA2311: DISCRETE MATHEMATICS

UNIT I LOGIC AND PROOFS

WEEK 1: Propositional Logic – Propositional equivalences

WEEK 2: Predicates and quantifiers-Nested Quantifiers

WEEK 3: Rules of inference-introduction to Proofs-Proof Methods and strategy

UNIT II COMBINATORICS

WEEK 4 : UNIT TEST-I

Mathematical inductions-Strong induction and well ordering-.The basics of counting-The pigeonhole principle –Permutations and combinations-Recurrence relations

WEEK 5: Permutations and combinations-Recurrence relations Solving Linear recurrence relations-generating functions-inclusion and exclusion and applications.

WEEK 6 :- UNIT TEST-II

WEEK 7 :- UNIT III GRAPHS

Graph terminology

WEEK 8: special types of graphs-Representing graphs

WEEK 9: graph isomorphism connectivity-Euler and Graphs and graph models

WEEK 10 : Hamilton paths

UNIT IV ALGEBRAIC STRUCTURES

WEEK 11: UNIT TEST-III

Algebraic systems-Semi groups and monoids-Groups-Subgroups and homomorphisms

WEEK 12: Cosets and Lagrange's theorem- Ring & Fields (Definitions and examples)

WEEK 13: UNIT TEST-IV

WEEK 14: UNIT V LATTICES AND BOOLEAN ALGEBRA

Partial ordering

WEEK 15: -Posets-Lattices as Posets- Properties of lattices

WEEK 16 : Lattices as Algebraic systems –Sub lattices

WEEK 17: direct product and Homomorphism-Some Special lattices-Boolean Algebra

WEEK 18:- UNIT TEST-V

WEEK 19: MODEL EXAM

TEXT BOOKS

- 1. Kenneth H.Rosen, “Discrete Mathematics and its Applications”, Special Indian edition, Tata McGraw-Hill Pub. Co. Ltd., New Delhi, (2007). (For the units 1 to 3, Sections 1.1 to 1.7 , 4.1 & 4.2, 5.1 to 5.3, 6.1, 6.2, 6.4 to 6.6, 8.1 to 8.5)**

2. Trembly J.P and Manohar R, “Discrete Mathematical Structures with Applications to Computer Science”, Tata McGraw–Hill Pub. Co. Ltd, New Delhi, 30th Re-print (2007).(For units 4 & 5 , Sections 2-3.8 & 2-3.9,3-1,3-2 & 3-5, 4-1 & 4-2)

REFERENCES:

1. Ralph. P. Grimaldi, “Discrete and Combinatorial Mathematics: An Applied Introduction”, Fourth Edition, Pearson Education Asia, Delhi, (2002).
2. Thomas Koshy, ”Discrete Mathematics with Applications”, Elsevier Publications, (2006).
3. Seymour Lipschutz and Mark Lipson, ”Discrete Mathematics”, Schaum’s Outlines, Tata McGraw-Hill Pub. Co. Ltd., New Delhi, Second edition, (2007)

CS2302 : COMPUTER NETWORKS

UNIT I DATA COMMUNICATIONS

WEEK 1:. Components – Direction of Data flow – networks – Components and Categories – types of Connections

WEEK 2: Topologies –Protocols and Standards – ISO / OSI model – Transmission Media

WEEK 3: Coaxial Cable – Fiber Optics – Line Coding – Modems – RS232 Interfacing sequences. Error – detection and correction

UNIT II DATA LINK LAYER

WEEK4:– UNIT TEST-I

Parity – LRC – CRC – Hamming code – low Control and Error control - stop and wait – go back-N ARQ

WEEK 5:_selective repeat ARQ- sliding window – HDLC. - LAN - Ethernet IEEE 802.3 - IEEE 802.4 - IEEE 802.5 - IEEE 802.11 – FDDI - SONET – Bridges.

WEEK 6: UNIT TEST-II

WEEK 7:

UNIT III NETWORK LAYER

Internetworks

WEEK 8: – Packet Switching and Datagram approach IP addressing methods – Subnetting

WEEK 9: Routing – Distance Vector Routing – Link State Routing

WEEK 10 : Routers.

UNIT IV TRANSPORT LAYER

WEEK 11: UNIT TEST-III

Duties of transport layer Multiplexing – Demultiplexing – Sockets – User Datagram Protocol (UDP)

WEEK 12: Transmission Control Protocol (TCP) – Congestion Control – Quality of services (QOS) – Integrated Services.

WEEK 13: UNIT TEST-IV

WEEK 14: UNIT V APPLICATION LAYER

WEEK 15: Domain Name Space (DNS) – SMTP – FTP

WEEK 16: HTTP - WWW – Security

WEEK 17 :- – Cryptography.

WEEK 18:- UNIT TEST-V

WEEK 19: MODEL EXAM

TEXT BOOKS

- 1. Behrouz A. Forouzan, “Data communication and Networking”, Tata McGraw-Hill, 2004.**

REFERENCES

- 1. James F. Kurose and Keith W. Ross, “Computer Networking: A Top-Down Approach Featuring the Internet”, Pearson Education, 2003.**
- 2. Larry L. Peterson and Peter S. Davie, “Computer Networks”, Harcourt Asia Pvt. Ltd., Second Edition.**
- 3. Andrew S. Tanenbaum, “Computer Networks”, PHI, Fourth Edition, 2003.**
- 4. William Stallings, “Data and Computer Communication”, Sixth Edition, Pearson Education, 2000.**

CS2303: THEORY OF COMPUTATION

UNIT I AUTOMATA

WEEK 1: Introduction to formal proof – Additional forms of proof – Inductive proofs – Finite Automata (FA)

WEEK 2: Deterministic Finite Automata (DFA)

WEEK 3: – Non-deterministic Finite Automata (NFA) – Finite Automata with Epsilon transitions

UNIT II REGULAR EXPRESSIONS AND LANGUAGES

WEEK 4: – UNIT TEST-I

Regular Expression-FA and Regular Expressions – Proving languages not to be regular

WEEK 5: – Closure properties of regular languages – Equivalence and minimization of Automata.

WEEK 6: UNIT TEST-II

WEEK 7:

UNIT III CONTEXT-FREE GRAMMARS AND LANGUAGES

Context-Free Grammar (CFG)

**WEEK 8:– Parse Trees - Ambiguity in grammars and languages
Definition of the Pushdown automata**

**WEEK 9: Languages of a Pushdown Automaton – Equivalence of
Pushdown automata and CFG**

WEEK 10: Deterministic Pushdown Automata

UNIT IV PROPERTIES OF CONTEXT-FREE LANGUAGES

WEEK 11: UNIT TEST-III

**Normal forms for CFG - – Pumping Lemma for CFL – Closure
Properties of CFL**

WEEK 12: Turing Machines – Programming Techniques for TM.

WEEK 13 : UNIT TEST-IV

WEEK 14:

UNIT V UNDECIDABILITY

**WEEK 15: A language that is not Recursively Enumerable (RE) –
An undecidable problem that is RE**

WEEK 16 : Undecidable problems about Turing Machine

**WEEK 17: Post’s Correspondence Problem – The classes P and
NP.**

WEEK 18:- UNIT TEST-V

WEEK 19: MODEL EXAM

TEXT BOOK

1. J.E. Hopcroft, R. Motwani and J.D. Ullman, “Introduction to Automata Theory, Languages and Computations”, second Edition, Pearson Education, 2007.

REFERENCES

1. H.R. Lewis and C.H. Papadimitriou, “Elements of the theory of Computation”, Second Edition, Pearson Education, 2003.
2. Thomas A. Sudkamp,” An Introduction to the Theory of Computer Science, Languages and Machines”, Third Edition, Pearson Education, 2007.
3. Raymond Greenlaw an H.James Hoover, “ Fundamentals of Theory of Computation, Principles and Practice”, Morgan Kaufmann Publishers, 1998.
- 4.Micheal Sipser, “Introduction of the Theory and Computation”, Thomson Brokecole, 1997.
5. J. Martin, “Introduction to Languages and the Theory of computation” Third Edition, Tata Mc Graw Hill, 2007

CS2306: SYSTEM SOFTWARE

UNIT I INTRODUCTION

WEEK 1: System software and machine architecture – The Simplified Instructional Computer (SIC)

WEEK 2: – Machine architecture - Data and instruction formats

WEEK 3: Addressing modes - instruction sets - I/O and programming.

UNIT II ASSEMBLERS

WEEK 4: UNIT TEST-I

Basic assembler functions - A simple SIC assembler – Assembler algorithm and data structures - Machine dependent assembler features - Instruction formats and addressing modes

WEEK 5:_ Program relocation - Machine independent assembler features - Literals – Symbol-defining statements – Expressions - One pass assemblers and Multi pass assemblers - Implementation example - MASM assembler.

WEEK 6: UNIT TEST-II

WEEK 7:

UNIT III LOADERS AND LINKERS

Basic loader functions - Design of an Absolute Loader – A Simple Bootstrap Loader - Machine dependent loader features

WEEK 8: Relocation – Program Linking – Algorithm and Data Structures for Linking Loader - Machine-independent loader features

WEEK 9: Automatic Library Search – Loader Options - Loader design options - Linkage Editors – Dynamic Linking – Bootstrap Loaders - Implementation example - MSDOS linker

WEEK 10: UNIT TEST-III

UNIT IV MACRO PROCESSORS

Basic macro processor functions - Macro Definition and Expansion

WEEK 11: – Macro Processor Algorithm and data structures - Machine-independent macro processor features - Concatenation of Macro Parameters

WEEK 12: Generation of Unique Labels – Conditional Macro Expansion – Keyword Macro Parameters-Macro within Macro-Implementation example - MASM Macro Processor – ANSI C Macro language.

WEEK 13 : UNIT TEST-IV

WEEK 14:

UNIT V SYSTEM SOFTWARE TOOLS

Text editors - Overview of the Editing Process

WEEK 15: - User Interface – Editor Structure. - Interactive debugging systems

WEEK 16: Debugging functions and capabilities – Relationship with other parts of the system – User-Interface Criteria.

WEEK 17: User-Interface Criteria

WEEK 18:- UNIT TEST-V

WEEK 19: MODEL EXAM

TEXT BOOK

1. Leland L. Beck, “System Software – An Introduction to Systems Programming”, 3rd Edition, Pearson Education Asia, 2006.

REFERENCES

1. D. M. Dhamdhere, “Systems Programming and Operating Systems”, Second Revised Edition, Tata McGraw-Hill, 2000.
2. John J. Donovan “Systems Programming”, Tata McGraw-Hill Edition, 2000.
3. John R. Levine, Linkers & Loaders – Harcourt India Pvt. Ltd., Morgan Kaufmann Publishers, 2000.

CS2304 : PROGRAMMING PARADIGMS

UNIT I OBJECT-ORIENTED PROGRAMMING – FUNDAMENTALS

WEEK 1: Review of OOP - Objects and classes in Java – defining classes

WEEK2: methods - access specifiers – static members – constructors – finalize method

WEEK 3: Arrays – Strings - Packages – JavaDoc comments

UNIT II OBJECT-ORIENTED PROGRAMMING **INHERITANCE**

WEEK 4: UNIT TEST-I

**Inheritance – class hierarchy – polymorphism -Dynamic binding
– final keyword – abstract classes – the Object class**

**WEEK 5: Reflection – interfaces – object cloning – inner classes –
proxies**

WEEK 6 : UNIT TEST-II

WEEK: 7:

UNIT III EVENT-DRIVEN PROGRAMMING

**Graphics programming – Frame – Components – working with
2D shapes**

**WEEK 8: Using color, fonts, and images - Basics of event
handling – event handlers – adapter classes**

**WEEK 9: actions – mouse events – AWT event hierarchy –
introduction to Swing – Model-View-Controller design pattern –
buttons – layout management – Swing Components**

UNIT IV GENERIC PROGRAMMING

WEEK10 : UNIT TEST-III

Motivation for generic programming – generic classes

**WEEK 11: generic methods – generic code and virtual machine –
inheritance and generics – reflection and generics**

**WEEK 12: exceptions – exception hierarchy – throwing and
catching exceptions – Stack Trace Elements - assertions – logging**

WEEK 13 :- UNIT TEST-IV

WEEK 14: REVISION UNIT- I-IV

UNIT V CONCURRENT PROGRAMMING

WEEK 15: Multi-threaded programming – interrupting threads – thread states – thread properties

WEEK 16:- thread synchronization – thread-safe Collections – Executors – synchronizers

WEEK 17 :-threads and event-driven programming

**WEEK 18:- UNIT TEST-I
MODEL PRACTICAL EXAM**

WEEK 19: MODEL EXAM

TEXT BOOK

1. Cay S. Horstmann and Gary Cornell, “Core Java: Volume I – Fundamentals”, Eighth Edition, Sun Microsystems Press, 2008.

REFERENCES

1. K. Arnold and J. Gosling, “The JAVA programming language”, Third edition, Pearson Education, 2000.
2. Timothy Budd, “Understanding Object-oriented programming with Java”, Updated Edition, Pearson Education, 2000.
3. C. Thomas Wu, “An introduction to Object-oriented programming with Java”, Fourth Edition, Tata McGraw-Hill Publishing company Ltd., 2006.

CS2305 : NETWORKS LAB

LIST OF EXPERIMENTS

- 1. Programs using TCP Sockets (like date and time server & client, echo server & client, etc..)**
- 2. Programs using UDP Sockets (like simple DNS)**
- 3. Programs using Raw sockets (like packet capturing and filtering)**
- 4. Programs using RPC**
- 5. Simulation of sliding window protocols**
- 6. Experiments using simulators (like OPNET)**
- 7. Performance comparison of MAC protocols**
- 8. Performance comparison of Routing protocols**
- 9. Study of TCP/UDP performance**

CS2307 : SYSTEM SOFTWARE LAB

LIST OF EXPERIMENTS

(Using C)

- 1. Implement a symbol table with functions to create, insert, modify, search, and display.**
- 2. Implement pass one of a two pass assembler.**
- 3. Implement pass two of a two pass assembler.**
- 4. Implement a single pass assembler.**
- 5. Implement a two pass macro processor**
- 6. Implement a single pass macro processor.**
- 7. Implement an absolute loader.**

- 8. Implement a relocating loader.**
- 9. Implement pass one of a direct-linking loader.**
- 10. Implement pass two of a direct-linking loader.**
- 11. Implement a simple text editor with features like insertion / deletion of a character, word, and sentence.**
- 12. Implement a symbol table with suitable hashing**

CS2306: JAVA PROGRAMMING LAB

- 1. Develop Rational number class in Java. Use JavaDoc comments for documentation. Your implementation should use efficient representation for a rational number, i.e. (500 / 1000) should be represented as $\frac{1}{2}$.**
- 2. Develop Date class in Java similar to the one available in java.util package. Use JavaDoc comments.**
- 3. Implement Lisp-like list in Java. Write basic operations such as 'car', 'cdr', and 'cons'. If L is a list [3, 0, 2, 5], L.car() returns 3, while L.cdr() returns [0,2,5].**
- 4. Design a Java interface for ADT Stack. Develop two different classes that implement this interface, one using array and the other using linked-list. Provide necessary exception handling in both the implementations.**
- 5. Design a Vehicle class hierarchy in Java. Write a test program to demonstrate polymorphism.**
- 6. Design classes for Currency, Rupee, and Dollar. Write a program that randomly generates Rupee and Dollar objects and write them into a file using object serialization. Write another**

program to read that file, convert to Rupee if it reads a Dollar, while leave the value as it is if it reads a Rupee.

7. Design a scientific calculator using event-driven programming paradigm of Java.

8. Write a multi-threaded Java program to print all numbers below 100,000 that are both prime and fibonacci number (some examples are 2, 3, 5, 13, etc.). Design a thread that generates prime numbers below 100,000 and writes them into a pipe. Design another thread that generates fibonacci numbers and writes them to another pipe. The main thread should read both the pipes to identify numbers common to both.

9. Develop a simple OPAC system for library using even-driven and concurrent programming paradigms of Java. Use JDBC to connect to a back-end database.

10. Develop multi-threaded echo server and a corresponding GUI client in Java.

11. [Mini-Project] Develop a programmer's editor in Java that supports syntax-highlighting, compilation support, debugging support, etc.