## Scheme of Teaching and Examination

### M.Tech. (Computer Technology)

#### Semester - I

<table>
<thead>
<tr>
<th>S.N.</th>
<th>Board of Study</th>
<th>Subject Code</th>
<th>Subject Name</th>
<th>Periods per week</th>
<th>Scheme of Exam</th>
<th>Total Marks</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
<td>ESE</td>
</tr>
<tr>
<td>1</td>
<td>Computer Sc. &amp; Engg.</td>
<td>554111(22)</td>
<td>Compiler Design</td>
<td>3</td>
<td>1</td>
<td></td>
<td>100</td>
</tr>
<tr>
<td>2</td>
<td>Computer Sc. &amp; Engg.</td>
<td>522112(22)</td>
<td>Java Programming &amp; Applications</td>
<td>3</td>
<td>1</td>
<td></td>
<td>100</td>
</tr>
<tr>
<td>3</td>
<td>Computer Sc. &amp; Engg.</td>
<td>522113(22)</td>
<td>Advanced Computer Architecture</td>
<td>3</td>
<td>1</td>
<td></td>
<td>100</td>
</tr>
<tr>
<td>4</td>
<td>Computer Sc. &amp; Engg.</td>
<td>554112(22)</td>
<td>Advanced Computer Communication Network</td>
<td>3</td>
<td>1</td>
<td></td>
<td>100</td>
</tr>
<tr>
<td>5</td>
<td>Computer Sc. &amp; Engg.</td>
<td>Elective –I</td>
<td></td>
<td>3</td>
<td>1</td>
<td></td>
<td>100</td>
</tr>
<tr>
<td>6</td>
<td>Computer Sc. &amp; Engg.</td>
<td>554121(22)</td>
<td>Compiler Design Lab</td>
<td>-</td>
<td>-</td>
<td>3</td>
<td>75</td>
</tr>
<tr>
<td>7</td>
<td>Computer Sc. &amp; Engg.</td>
<td>554122(22)</td>
<td>Java Programming &amp; Applications Lab</td>
<td>-</td>
<td>-</td>
<td>3</td>
<td>75</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td></td>
<td></td>
<td>15</td>
<td>5</td>
<td>6</td>
<td>650</td>
</tr>
</tbody>
</table>

L- Lecture, T- Tutorial, P - Practical, ESE- End Semester Examination, CT- Class Test, TA- Teacher's Assessment

Note : Duration of all theory papers will be of Three Hours.

#### Elective –I

<table>
<thead>
<tr>
<th>S.N.</th>
<th>Board of Study</th>
<th>Code</th>
<th>Subject Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Computer Science &amp; Engineering</td>
<td>522133(22)</td>
<td>System Simulation</td>
</tr>
<tr>
<td>2</td>
<td>Computer Science &amp; Engineering</td>
<td>522134(22)</td>
<td>Analysis and Design of Algorithms</td>
</tr>
<tr>
<td>3</td>
<td>Computer Science &amp; Engineering</td>
<td>522135(22)</td>
<td>Digital Signal Processing</td>
</tr>
</tbody>
</table>
Unit - I
Introduction
Compilers – Analysis of the source program, Phases of a compiler, Cousins of the Compiler, Grouping of Phases, Compiler construction tools; Lexical Analysis – Role of Lexical Analyzer, Input Buffering, Specification of Tokens. Lexical phase errors, syntactic phase errors, semantic phase errors.

Unit – II
Syntax Analysis and Parsing Techniques
Role of the parser, Writing Grammars – Context-Free Grammars, Top Down parsing, Recursive Descent Parsing, Predictive Parsing, Bottom-up parsing, Shift Reduce Parsing, Operator Precedent Parsing, LR Parsers, SLR Parser, Canonical LR Parser, LALR Parser.

Unit - III
Intermediate Code Generation & Syntax Directed Translation

Unit - IV
Code Generation
Issues in the design of code generator, Target Machine, Runtime Storage Management, Basic Blocks and Flow Graphs, Next-Use Information, A Simple Code Generator, DAG Representation of Basic Blocks, Peephole Optimization.

Unit - V
Code Optimization and Run Time Environments

Text Books

References Books
Unit-I

Introduction to Java

Importance and Features of Java, Concepts of Java Virtual machine (JVM), Keywords, Constants, Variables and Data types, Operators and Expressions, Control Statements, Conditional Statements, Loops and Iterations. Class Definition, Adding Variables and Methods, Creating Objects, Constructors, Defining methods, Calling methods, Method overloading. Creating an Array, One and Two Dimensional Array, String Array and Methods String and String Buffer classes, Wrapper Classes.

Unit-II

Inheritance

Basic Types, Super classes, Multilevel Hierarchy Abstract and Final classes, Object Class, Packages and Interfaces, Access protection, extending Interfaces, Exception Handling, Fundamental Exception types, Uncaught Exception, Throw, Throws, final Methods, Creating own Exceptions

Unit-III

Multithreaded programming


Unit-IV

Input/output

Basics, Streams, Byte and Character Streams, Predefined Streams, Reading and Writing from Console and Files, using Java Packages.

Networking in Java

Networking fundamentals, Client/server model, Internet addresses, Sockets, networking classes and Interfaces, using Java.net Package, TCP/IP and Datagram Programming, HTTP Protocol and URLs.

Unit-V

Event Handling

Different Mechanism, the Delegation Event Model, Classes, Event Listener Interfaces, Adapter and Inner Classes, Working with Windows, Graphics and Text, using AWT Controls, Layout Managers and Menus, Handling Image, Animation, Sound and Video Java Applet, Introduction to Swings, JDBC and Servlets.

Programming Graphical System:

Text Books


**Reference Books**

CHHATTISGARH SWAMI VIVEKANAND TECHNICAL UNIVERSITY, BHILAI

Subject: Advanced Computer Architecture Code: 522113(22)
Total Theory Periods: 40 Total Tut. Periods: 12
Total Marks in End Semester Exam: 100 Minimum number of Class Tests to be conducted: 02

Unit - I

UNIT - II
Advanced Processor Technology- CISC, RISC, Superscalar, Vector, VLIW and Symbolic Processors, Memory Hierarchy Technology, Virtual Memory Technology (Virtual memory Models, TLB, Paging and Segmentation).

UNIT - III
Cache Memory Organization, Shared Memory Organization, Sequential and Weak Consistency Models

UNIT - IV
Linear Pipeline Processors, Nonlinear Pipeline processors, Instruction Pipeline Design, Arithmetic Pipeline Design

UNIT - V

Text Book

Reference Books
Unit -I
Introduction

Unit- II
Data Communications

Unit -III
Network Layer

UNIT IV
TRANSPORT LAYER

UNIT V
APPLICATION LAYER

Text Books

Reference Book
Unit - I


Unit-II


Unit-III


Unit-IV


Unit-V

Simulation of a General Inventory System, Simulation of an Inventory Policy (P, Q), Simulation of an Inventory System with Buffer Stock, Simulation Languages.

Text Books

Reference Books
Unit -I

Unit -II
Divide and Conquer technique of problem solving, sorting algorithms: Quicksort, Merge Sort, Merging Sorted sequences, Lower brands for sorting, heap sort, shell sort, radix sort, Dynamic sets and searching : Array doubling, Red Black trees, hashing high, priority queues.

Unit -III
Graphs: Definitions and representations, traversal, DFS and BFS., DFS on undirected graphs.

Unit- IV
Dynamic Programming: Sub problem, Graphs and their Traversal, Multiplying a Sequence of matrices, Optimal Binary Search Tree Construction.

Unit-V

Text Book :

References :
Unit-I

Unit - II

Unit - III

Unit - IV

Unit - V
Text Books

Reference Books
Experiments to be performed

1. WAP to generate tokens in give source language using in C.
2. Extend program 1 to check whether a given token is a key word or a data type. If it is data type check whether it is valid data type.
3. Extend program 2 to implement a lexical analyzer in C.
4. Use LEX tool to implement a lexical analyzer.
5. Develop an operator precedence a parser (construct a parse table algo)/ LL(1) parser.
6. Implement a recursive descent parser for an expression grammar that generates arithmetic expressions with digits, + and *.
7. Use YACC and LEX to implement a parser for the same grammar as given in problem
8. Write semantic rules to the YACC program in problem 5 and implement a calculator that takes an expression with digits, + and * and computes and prints its value.
9. Implement the front end of a compiler that generates the three address code for a simple language with: one data type integer, arithmetic operators, relational operators, variable declaration statement, one conditional construct, one iterative construct and assignment statement.
10. Implement the back end of the compiler which takes the three address code generated in problems 7 and 8, and produces the 8086 assembly language instructions that can be assembled and run using an 8086 assembler. The target assemblies instructions can be simple move, add, sub, and jump. Also simple addressing modes are used.
Experiments to be performed

1. Write a program for matrix multiplication. Use InputStream Reader and Buffered Reader classes for Input/Output.
2. Write a program to create a user defined Exception when the user inputs the marks which exceed more than 100.
3. Write a program to animate a string on Applet. Use the concept of Multithreading.
4. Write a program to design a calculator using the AWT controls provided in Java.
5. Write a program for Client Server communication using either UDP or TCP protocols. Use Server Socket and Socket classes.
6. Write a program to create some of the features of Notepad. Use Swings for designing this application.
7. Create functions like multiply, addition and subtraction respectively. Invoke these functions from remote system by using the concept of Remote Method Invocation in Java.
8. Create a form containing fields name and password respectively, using applet as a container
9. The input entered in these fields should be stored in the database. Use JDBC connectivity for implementing this program.
10. Write a program to create a small portal which contains the registration form of students. Use JDBC.
11. Write a program create a bean component in Java for addition of two numbers.