ST.JOSEPH’S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM

IV SEMESTER **COMPUTER SCIENCE**  TIME: 4HRS/WEEK

CS4603(3) **OBJECT ORIENTED PROGRAMMING THROUGH JAVA** MAX.MARKS:100

w.e.f. 20-21 admitted batch-“20AH” **SYLLABUS**

**Course Objectives:**

* Gain knowledge about basic Java language syntax and semantics to write Java programs and use concepts such as variables, conditional and iterative execution methods.
* Understand the fundamentals of object-oriented programming in Java, including defining classes, objects, invoking methods etc and exception handling mechanisms.
* Understand the principles of inheritance, packages and interfaces.

**Course Outcomes :**

* To recognize how to execute a simple as well as Java application and underlying the principles of Object-Oriented Programming.
* Describe and implement various Inheritance and Polymorphism forms using Java Classes and Interfaces.
* Implement efficient Java applets, exception handling and multithreading concepts in real life programming domains and hence enhance employability skills.

**UNIT - I: INTRODUCTION TO JAVA:** Features of Java, The Java virtual Machine, Parts of Java.

**NAMING CONVENTIONS AND DATA TYPES:** Naming Conventions in Java, Data Types in Java, Literals

**OPERATORS IN JAVA:** Operators, Priority of Operators.

**CONTROL STATEMENTS IN JAVA:** if... else Statement, do... while Statement, while Loop, for Loop, switch Statement, break Statement, continue Statement, return Statement

**INPUT AND OUTPUT:** Accepting Input from the Keyboard, Reading Input with Java. util. Scanner Class, Displaying Output with System. out. printf(), Displaying Formatted Output with String. format()

**ARRAYS:** Types of Arrays, Three Dimensional Arrays (3D array), array name. length, Command Line Arguments

**UNIT-II: STRINGS:** Creating Strings, String Class Methods, String Comparison, Immutability of Strings

**INTRODUCTION TO OOPS:** Problems in Procedure Oriented Approach, Features of Object-Oriented Programming System (OOPS)

**CLASSES AND OBJECTS:** Object Creation, Initializing the Instance Variables, Access Specifies, Constructors

**METHODS IN JAVA:** Method Header or Method Prototype, Method Body, Understanding Methods, Static Methods, Static Block, The keyword ‘this’, Instance Methods, Passing Primitive Data Types to Methods, Passing Objects to Methods, Passing Arrays to Methods, Recursion, Factory Methods.

**INHERITANCE:** Inheritance, The keyword ‘super’, The Protected Specifier, Types of Inheritance

**UNIT-III: POLYMORPHISM:** Polymorphism with Variables, Polymorphism using Methods, Polymorphism with Static Methods, Polymorphism with Private Methods, Polymorphism with Final Methods, final Class

**TYPE CASTING:** Types of Data Types, Casting Primitive Data Types, Casting Referenced Data Types, the Object Class

**ABSTRACT CLASSES:** Abstract Method and Abstract Class

**INTERFACES:** Interface, Multiple Inheritance using Interfaces

**PACKAGES:** Package, Different Types of Packages, The JAR Files, Interfaces in a Package, Creating Sub Package in a Package, Access Specifiers in Java, Creating API Document

**EXCEPTION HANDLING:** Errors in Java Program, Exceptions, throws Clause, throw Clause, Types of Exceptions, Re – throwing an Exception

**UNIT-IV: STREAMS:** Stream, Creating a File using File Output Stream, Reading Data from a File using File Input Stream, Creating a File using File Writer, Reading a File using File Reader, Zipping and Unzipping Files, Serialization of Objects, Counting Number of Characters in a File, File Copy, File Class.

**THREADS:** Single Tasking, Multi Tasking, Uses of Threads, Creating a Thread and Running it, Terminating the Thread, Single Tasking Using a Thread, Multi Tasking Using Threads, Multiple Threads Acting on Single Object, Thread Class Methods, Deadlock of Threads, Thread Communication, Thread Priorities, thread Group, Daemon Threads, Applications of Threads, Thread Life Cycle.

**UNIT V : APPLETS:** Creating an Applet, Uses of Applets, <APPLET> tag, A Simple Applet, An Applet with Swing Components, Animation in Applets, A Simple Game with an Applet, Applet Parameters

**JAVA DATABASE CONNECTIVITY:** Database Servers, Database Clients, JDBC (Java Database Connectivity), Working with Oracle Database, Working with MySQL Database, Stages in a JDBC Program, Registering the Driver, Connecting to a Database, Preparing SQL Statements, Using jdbc–odbc Bridge Driver to Connect to Oracle Database, Retrieving Data from MySQL Database, Retrieving Data from MS Access Database, Stored Procedures and Callable Statements, Types of Result Sets

**PRESCRIBED TEXT BOOK:** Core Java: An Integrated Approach, Authored by Dr. R. Nageswara Rao & Kogent Learning Solutions Inc.

**REFERENCE BOOKS:**

* E.Balaguruswamy, Programming with JAVA, A primer, 3e, TATA McGraw- Hill Company.
* John R. Hubbard, Programming with Java, Second Edition, Schaum’s outline Series, TMH.
* Deitel&amp;Deitel. Java TM: How to Program, PHI (2007).

\*\* \*\* \*\*

ST.JOSEPH’S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM

IV SEMESTER **COMPUTER SCIENCE**  TIME: 2HRS/WEEK

CS 4653 (2) **OBJECT ORIENTED PROGRAMMING THROUGH JAVA LAB** MAX.MARKS:50

w.e.f. 20-21 admitted batch-“20AH” **PRACTICALS** **SYLLABUS**

**COURSE OBJECTIVES:** To enable the students to:

* Understand the fundamentals of object oriented programming in java, including defining classes, invoking objects along with constructors, arrays and vectors.
* Discuss the principles of inheritance, interface and packages.

**COURSE OUTCOMES:**

After Completion of this course the student would be able to:

* Use an integrated development environment to write, compile, run and test simple object oriented java programs.
* Apply skills using basic control structures, arrays, object oriented principles including encapsulation and information hiding.
* Implement multithreaded programs and Exception handling.
* Apply the programming concepts as and when required in the future application development.

**1.** Write a program to read ***Student Name, Reg.No, Marks[5]*** and calculate ***Total***, ***Percentage, Result***. Display all the details of students

**2.** Write a program to perform the following String Operations

**a.** Read a string

**b.** Find out whether there is a given substring or not

**c.** Compare existing string by another string and display status

**d.** Replace existing string character with another character

**e.** Count number of works in a string

**3.** Java program to implements Addition and Multiplication of two N X N matrices.

**4.** Java program to demonstrate the use of Constructor.

**5.** Calculate area of the following shapes using method overloading.

**a.** Triangle

**b.** Rectangle

**c.** Circle

**d.** Square

**6.** Implement inheritance between ***Person (Aadhar, Surname, Name, DOB, and Age)*** and ***Student (Admission Number, College, Course, Year)***classes where ReadData(), DisplayData() are overriding methods.

**7.** Java program for implementing Interfaces

**8.** Java program on Multiple Inheritance.

**9.** Java program for to display ***Serial Number from 1 to N*** by creating two Threads

**10.** Java program to demonstrate the following exception handlings

a. Divided by Zero

b. Array Index Out of Bound

c. File Not Found

d. Arithmetic Exception

e. User Defined Exception

**11.** Create an Applet to display different shapes such as Circle, Oval, Rectangle, Square and Triangle.

12. Write a program to create ***Book (ISBN,Title, Author, Price, Pages, Publisher)***structure and store book details in a file and perform the following operations

a. Add book details

b. Search a book details for a given ISBN and display book details, if available

c. Update a book details using ISBN

d. Delete book details for a given ISBN and display list of remaining Books

\*\* \*\* \*\*

ST.JOSEPH’S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM

IV SEMESTER **COMPUTER SCIENCE**  Time: 4Hrs/Week

**OPERATING SYSTEMS**

20-21 admitted batch-“20AH” **SYLLABUS** Max.Marks:100

**COURSE OBJECTIVES:**

To enable the students to:

* Understand the overall structure and components of operating system.
* Analyze the key concept of Process Management and concurrency problem.
* Understand different approaches to memory management.

**COURSE OUTCOMES:**

Upon successful completion of the course, a student will be able to:

* Demonstrate the structure and design of operating systems.
* **Compare various algorithms for process scheduling.**
* **Apply various deadlock handling strategies to solve resource allocation problems.**
* **Evaluate the performance of different memory management techniques and page replacement algorithms and therefore develop employability skills.**
* Describe file concepts and analyse various disk scheduling strategies.

**UNIT- I**

What is Operating System? History and Evolution of OS, Basic OS functions, Resource Abstraction, Types of Operating Systems– Multiprogramming Systems, Batch Systems, Time Sharing Systems; Operating Systems for Personal Computers, Workstations and Hand-held Devices, Process Control & Real time Systems.

**UNIT- II**

Processor and User Modes, Kernels, System Calls and System Programs, System View of the Process and Resources, Process Abstraction, Process Hierarchy, Threads, Threading Issues, Thread Libraries; Process Scheduling, Non-Preemptive and Preemptive Scheduling Algorithms.

**UNIT III**

**Process Management:** Deadlock, Deadlock Characterization, Necessary and Sufficient Conditions for Deadlock, Deadlock Handling Approaches: Deadlock Prevention, Deadlock Avoidance and Deadlock Detection and Recovery.

Concurrent and Dependent Processes, Critical Section, Semaphores, Methods for Inter-process Communication; Process Synchronization, Classical Process Synchronization Problems: Producer-Consumer, Reader-Writer.

**UNIT IV**

**Memory Management:** Physical and Virtual Address Space; Memory Allocation Strategies– Fixed and -Variable Partitions, Paging, Segmentation, Virtual Memory.

**UNIT V**

**File and I/O Management, OS security** : Directory Structure, File Operations, File Allocation Methods, Device Management, Pipes, Buffer, Shared Memory, Security Policy Mechanism, Protection, Authentication and Internal Access Authorization

Introduction to Android Operating System, Android Development Framework, Android Application Architecture, Android Process Management and File System, Small Application Development using Android Development Framework.

<https://nptel.ac.in/courses/106/105/106105214/>

<http://www.infocobuild.com/education/audio-video-courses/computer-science/OperatingSystems-IIT-Delhi/lecture-36.html>

<https://www.youtube.com/watch?v=AnGOeYJCv6s>

<https://www.youtube.com/watch?v=U1Jpvni0Aak>

<https://nptel.ac.in/content/storage2/courses/126104006/LectureNotes/Week-2_IntroductionToAndroid.pdf>

<https://www.youtube.com/watch?v=fzQcQV0UCUM>

**Prescribed Text Book:**

Operating System Principles by Abraham Silberschatz, Peter Baer Galvin and Greg Gagne (7thEdition) Wiley India Edition.

**Reference Books:**

Operating Systems: Internals and Design Principles by Stallings (Pearson)

Operating Systems by J. Archer Harris (Author), Jyoti Singh (Author) (TMH)

Online Resources for UNIT V

ST.JOSEPH’S COLLEGE FOR WOMEN (AUTONOMOUS), VISAKHAPATNAM

IV SEMESTER **COMPUTER SCIENCE**  Time: 2Hrs/Week

**OPERATING SYSTEMS LAB USING C**

20-21 admitted batch-“20AH” **SYLLABUS** Max.Marks:50

**COURSE OBJECTIVES:**

To enable the students to:

* Analyze the concept of Process Management and concurrency problem.
* Understand different approaches to memory management.

**COURSE OUTCOMES:**

Upon successful completion of the course, a student will be able to:

* Evaluate the performance of different types of CPU scheduling algorithms
* Compare different page replacement policies.
* Implement file organization techniques.
* Recognize need of Banker’s algorithm for deadlock avoidance

1. Write a program to implement Round Robin CPU Scheduling algorithm

2. Simulate SJF CPU Scheduling algorithm

3. Write a program the FCFS CPU Scheduling algorithm

4. Write a program to Priority CPU Scheduling algorithm

5. Simulate Sequential file allocation strategies

6. Simulate Indexed file allocation strategies

7. Simulate Linked file allocation strategies

8. Simulate MVT and MFT memory management techniques

9. Simulate Single level directory File organization techniques

10. Simulate Two level File organization techniques

11. Simulate Hierarchical File organization techniques

12. Write a program for Bankers Algorithm for Dead Lock Avoidance

13. Implement Bankers Algorithm Dead Lock Prevention.

14. Simulate all Page replacement algorithms.

a) FIFO b) LRU c) LFU

15. Simulate Paging Technique of memory management