PARUL UNIVERSITY - Faculty of IT & Computer Science

Department of Computer Application

SYLLABUS FOR 1st Sem MCA, M.Sc. (IT) PROGRAMME

Java Programming (05201201)

Type of Course: MCA, M.Sc. (IT)

Prerequisite: Knowledge of C, C++

Rationale: To acquire the fundamental knowledge of Java programming

Teaching and Examination Scheme:

Teaching Scheme			Credit	Examination Scheme					Total
				External		Internal			
Lect Hrs/ Week	Tut Hrs/ Week	Lab Hrs/ Week		T	P	T	CE	P	
3	0	2	4	60	30	20	20	20	150

Lect - Lecture, **Tut** - Tutorial, **Lab** - Lab, **T** - Theory, **P** - Practical, **CE** - CE, **T** - Theory, **P** - Practical

Contents:

Sr.	Topic	Weightage	Teaching Hrs.
1	**Introduction to Java Data Types, Operators, Statements**: Paradigms of programming languages, Evolution of OO methodology, Basic concepts of OO approach, Comparison of object oriented and procedure oriented approaches. Concepts of OOP - classes and objects, abstraction and encapsulation, inheritance, polymorphism. Features of the Java language, Java environment, Object oriented programming in Java, Java program structure, Java and unicode, Data types, Variables and arrays - data types in Java, literals, characters, variable declaration, symbolic constants. Type casting operations in Java, Arithmetic operators, Basic assignment operators, Relational operators, Boolean logical operators, Ternary operator, Operator precedence. Control statements - Java's selection statements, switch, nested switch, iteration constructs, continue, return.	20%	8
2	**Inheritance, Sub Classing, Package**: Inheritance Concepts – defining sub classes, method overriding, using super keyword, Variable shadowing, Method and variable binding, Using final keyword, Abstract classes and interfaces. Object class, Packages - Creating package, CLASSPATH environment variable, access specifiers, Access Control / visibility.	20%	8

3	**Exception, Collection Frameworks**: Exception handling - types of exceptions, Throwable class, Keywords - try, catch, throw, throws and finally, Nested try statements, Java built in exceptions, User defined exceptions. Collection framework – Collections, List, Set, Enumeration, Iterator, ArrayList.	15%	8
4	**IO and Multi Threading**: java.io- File class, Creating directory, Input/Output basics, Streams (byte and character), Reading from and writing to console, Reading and writing files, PrintWriter Class, the transient modifier, RandomAccessFile, Introduction to multi-threading, Thread class and execution of thread, Runnable interface, ThreadGroup, Daemon threads, Thread states.	15%	8
5	**GUI Programming and Applets**: Introduction to AWT and Swing, Fundamentals of applets, Applet class, Applet life cycle, A simple banner applet, getDocumentBase(), getCodeBase(), showDocument(), AppletContext and AppletStub interface, Working with frames, windows, graphics, color, fonts. AWT controls - buttons, checkbox, choice, list and textField. Layout Managers - Flow Layout, Grid Layout and Border Layout. User interface events - event classes and event listener interfaces, Adapter classes.	30%	13

***Continuous Evaluation:**

It consists of Assignments/Seminars/Presentations/Quizzes/Surprise Tests (Summative/MCQ) etc.

Reference Books:

1. The Complete Reference Java J2SE (TextBook)
   Herbert Schildt; TMH Publishing Company Ltd; 5th Edition
2. Core Java Volume 1
   Cay Horstmann and Gary Cornell; Pearson Education; 8th Edition
3. The class of JAVA
   Pravin Jain; Pearson

Useful Links:

https://sites.google.com/a/paruluniversity.ac.in/java_div1/

Course Outcome:

After Learning the course the students shall be able to:

1. implement object oriented principles using Java.
2. identify errors and implement exception handling mechanism.
3. develop programs using multi-threading concepts.
4. manage source code in Java packages.
5. design and develop java applications using Applets, Abstract Window Toolkit and Swing API.

List of Practical:

1. **Basic datatype and looping**

Write a program for swapping and find a factorial value. Perform swapping without using third variable.

2. **Looping and Control structure**

   Write a program to accept a number from the user through command line and display whether the given number is palindrome or not.

3. **Array**

   Write a program to accept an array of integer from the user through command line and find prime numbers from the array.

4. **Class**

   Create a class Stack that defines an integer stack that can hold 10 values. Perform push and pop actions in a stack.

5. **Inheritance**

   Write a program to create a class Publisher with attributes publisher name and publisher id. Derive a subclass Book with attributes bookname, bookid and author name. All these data should be entered by the user. Create two methods getdata() and showdata() to display the details of book and publisher.

6. **Method Overloading**

   Write a program to create a class with two methods with same name addfunc(), one accepting two integer parameters and other accepting two double parameters. When method is called, the appropriate method should be selected depending on parameters passed(method overloading).

7. **Super and this keyword**

   Declare a variable called x with integer as the data type in base class and subclass. Make a method named as show() which displays the value of x in the superclass and subclass.

8. **Final class, abstract class and interface**

   Write a program to calculate the area, circumference and volume for all shapes. [Perform this application using final class, abstract class and interface]

9. **Exception handling**

   Write a program to enter two integers from the command line and display the division of those two numbers. Handle all the exceptions (i.e. ArrayIndexOutOfBoundsException, NumberFormatException, ArithmeticException) for invalid arguments passed.

10. **File Handling**

    Write a program to perform following actions and store output in file:

    1. Accept strings from user, convert it into uppercase and store it in a file.
    2. Write double value to a text file and also display the date on which the application was run inside the file.
    3. Delete a given file or directory and display appropriate message

11. **Collection Framework**

Write a program for creating an ArrayList, add the elements in array list and then obtain an array from ArrayList and display the contents and sum of those numbers.

12. Collection Framework

Create a class named Address and define name, city and state as the data members of this class. Create another class named Maillist and add the elements to the linked list and display the contents using Iterator interface.

13. Calendar class

Create a Gregorian Calendar. Display current date and time in default locale and time zone. Find out whether the current year is a leap year or not.

14. Regular expression

Write a program to find integers and decimal value from the string by using a pattern of regular expression

15. Thread and Runnable interface

Write a multi-threaded program which sets the priority of threads and gets the name of threads.

16. Thread (sleep method)

Write a program to calculate sum and factorial of numbers using sleep method.

17. Multi-threading

Write a program to create multi-threaded application to perform banking tasks. [For Example: Withdrawal and Deposit from Joint account.]

18. AWT and Swing

Write a program to create a Menubar and create 2 Menus File and Edit. Involve New, Open, Close as items in the File menu and then add a separator, then further add Save, Save As and again add separator and add another menu item named Print. Add all these items in File menu. Add Line, Rectangle and Circle as the menu items and add Radio Button before these menu items and add a separator. Then add Red, Green and Blue as menu items and add Check Box before these menu items.

19. Applet and AWT

Create Java Applets to perform following tasks:

1. To display simple calculator
2. To write the content of the text area in the file, whose name is given in the text box using Frame
3. To draw rectangle to the applet when mouse is dragged.

20. AWT and Swing

Write a program to create 3 radio buttons named C++, Java and Pascal and add on JPanel. The user is asked to identify which of them is not an OOP language. When the user selects on choice, the program responds by displaying whether the selection is correct or wrong.