CS405PC: JAVA PROGRAMMING

B.Tech. II Year II Semester								
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
CS405PC	**PCC**	3	1	0	4	30	70	100
Contact Classes: 45	**Tutorial Classes: 15**	**Practical Classes: Nil**				**Total Classes: 60**		

Prerequisite:

Course Objectives:

- To introduce the object oriented programmingconcepts.
- To understand object oriented programming concepts, and apply them in solving problems.
- To introduce the principles of inheritance and polymorphism; and demonstrate how they relate to the design of abstractclasses
- To introduce the implementation of packages andinterfaces
- To introduce the concepts of exception handling andmultithreading.
- To introduce the design of Graphical User Interface using applets and swing controls.

Course Outcomes:

- Able to solve real world problems using OOPs techniques.
- Able to understand the use of abstractclasses.
- Able to solve problems using java collection framework and I/O classes.
- Able to develop multithreaded applications withsynchronization.
- Able to develop applets for webapplications.
- Able to design GUI based applications

Unit - 1	**Object-Oriented Thinking**	No. of Classes:12

Object-Oriented Thinking- A way of viewing world – Agents and Communities, messages and methods, Responsibilities, Classes and Instances, Class Hierarchies- Inheritance, Method binding, Overriding and Exceptions, Summary of Object-Oriented concepts. Java buzzwords, An Overview of Java, Data types, Variables and Arrays, operators, expressions, control statements, Introducing classes, Methods and Classes, String handling.

Inheritance– Inheritance concept, Inheritance basics, Member access, Constructors, Creating Multilevel hierarchy, super uses, using final with inheritance, Polymorphism-ad hoc polymorphism, pure polymorphism, method overriding, abstract classes, Object class, forms of inheritance- specialization, specification, construction, extension, limitation, combination, benefits of inheritance, costs of inheritance.

Unit - 2	**Packages**	No. of Classes:12

Packages- Defining a Package, CLASSPATH, Access protection, importing packages.

Interfaces- defining an interface, implementing interfaces, Nested interfaces, applying interfaces, variables in interfaces and extending interfaces.

Stream based I/O (java.io) – The Stream classes-Byte streams and Character streams, Reading console Input and Writing Console Output, File class, Reading and writing Files, Random access file operations, The Console class, Serialization, Enumerations, auto boxing, generics.

Unit - 3	Exception handling	No. of Classes:12

Exception handling - Fundamentals of exception handling, Exception types, Termination or resumptive models, Uncaught exceptions, using try and catch, multiple catch clauses, nested try statements, throw, throws and finally, built- in exceptions, creating own exception sub classes.

Multithreading- Differences between thread-based multitasking and process-based multitasking, Java thread model, creating threads, thread priorities, synchronizing threads, inter thread communication.

Unit - 4	Collections Framework	No. of Classes:12

The Collections Framework (java.util)- Collections overview, Collection Interfaces, The Collection classes- Array List, Linked List, Hash Set, Tree Set, Priority Queue, Array Deque. Accessing a Collection via an Iterator, Using an Iterator, The For-Each alternative, Map Interfaces and Classes, Comparators, Collection algorithms, Arrays, The Legacy Classes and Interfaces- Dictionary, Hashtable ,Properties, Stack, Vector More Utility classes, String Tokenizer, Bit Set, Date, Calendar, Random, Formatter, Scanner

Unit - 5	GUI Programming	No. of Classes:12

GUI Programming with Swing – Introduction, limitations of AWT, MVC architecture, components, containers. Understanding Layout Managers, Flow Layout, Border Layout, Grid Layout, Card Layout, Grid Bag Layout.

Event Handling- The Delegation event model- Events, Event sources, Event Listeners, Event classes, Handling mouse and keyboard events, Adapter classes, Inner classes, Anonymous Inner classes.

A Simple Swing Application, Applets – Applets and HTML, Security Issues, Applets and Applications, passing parameters to applets. Creating a Swing Applet, Painting in Swing, A Paint example, Exploring Swing Controls- JLabel and Image Icon, JText Field, **The Swing Buttons**- JButton, JToggle Button, JCheck Box, JRadio Button, JTabbed Pane, JScroll Pane, JList, JCombo Box, Swing Menus, Dialogs.

Text Books:

1. Java The complete reference, 9thedition, Herbert Schildt, McGraw Hill Education (India) Pvt. Ltd.
2. Understanding Object-Oriented Programming with Java, updated edition, T. Budd, Pearson Education.

Reference Books:

1. An Introduction to programming and OO design using Java, J. Nino and F.A. Hosch, John Wiley &sons
2. Introduction to Java programming, Y. Daniel Liang, PearsonEducation.
3. Object Oriented Programming through Java, P. Radha Krishna, UniversityPress.
4. Programming in Java, S. Malhotra, S. Chudhary, 2ndedition, Oxford Univ.Press.
5. Java Programming and Object-oriented Application Development, R. A. Johnson, Cengage Learning.

Web References:

4. https://nptel.ac.in/courses/106/105/106105191/

CS408PC: JAVA PROGRAMMING LAB

B.Tech. II Year II Semester								
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
CS408PC	**PCC**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		0	0	2	1	30	70	100
Contact Classes: 0	**Tutorial Classes: 0**	**Practical Classes:30**				**Total Classes:30**		

Prerequisite: Nil

Course Objectives:

- To write programs using abstractclasses.
- To write programs for solving real world problems using java collection framework.
- To write multithreadedprograms.
- To write GUI programs using swing controls inJava.
- To introduce java compiler and eclipseplatform.
- To impart hands on experience with javaprogramming.

Course Outcomes:

- Able to write programs for solving real world problems using java collection framework.
- Able to write programs using abstractclasses.
- Able to write multithreadedprograms.
- Able to write GUI programs using swing controls in Java.

List of Experiments:

1. Use Eclipse or Net bean platform and acquaint with the various menus. Create a test project, add a test class, and run it. See how you can use auto suggestions, auto fill. Try code formatter and code refactoring like renaming variables, methods, and classes. Try debug step by step with a small program of about 10 to 15 lines which contains at least one if else condition and a forloop.

2. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result. Handle any possible exceptions like divided byzero.

3. a)Develop an applet in Java that displays a simplemessage.
   b) Develop an applet in Java that receives an integer in one text field, and computes its factorial Value and returns it in another text field, when the button named "Compute" isclicked.

4. Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num 2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception. Display the exception in a message dialogbox.

5. Write a Java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third

thread will print the value of cube of thenumber.

6. Write a Java program for the following: Create a doubly linked list of elements. Delete a given element from the abovelist.Display the contents of the list after deletion.

7. Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with "Stop" or "Ready" or "Go" should appear above the buttons in selected color. Initially, there is no messageshown.

8. Write a Java program to create an abstract class named Shape that contains two integers and an empty method named print Area (). Provide three classes named Rectangle, Triangle, and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the givenshape.

9. Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using Labels in GridLayout.

10. Write a Java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired (Use Adapterclasses).

11. Write a Java program that loads names and phone numbers from a text file where the data is organized as one line per record and each field in a record are separated by a tab (\t). It takes a name or phone number as input and prints the corresponding other value from the hash table (hint: use hash tables).

12. Write a Java program that correctly implements the producer – consumer problem using the concept of interth read communication.

13. Write a Java program to list all the files in a directory including the files present in all its subdirectories.

14. Write a Java program that implements Quick sort algorithm for sorting a list of names in ascendingorder

15. Write a Java program that implements Bubble sort algorithm for sorting in descending order and also shows the number of interchanges occurred for the given set ofintegers.

List of Equipment/Software (with Specifications or Range) Required:

- Ubuntu System
- Eclipse or Net bean

References

1. Java for Programmers, P. J. Deitel and H. M. Deitel, 10th Edition *Pearson* education.
2. Thinking in Java, Bruce Eckel, *Pearson* Education.
3. Java Programming, D. S. Malik and P. S. Nair, *Cengage* Learning.
4. Core Java, Volume 1, 9thedition, Cay S. Horstmann and G Cornell,*Pearson*.