UNIVERSITY OF PETROLEUM & ENERGY STUDIES

College of Engineering Studies

Dehradun

COURSE PLAN

Programme   : B. Tech in (CSE+ All IBM Branches)

Course   : Object Oriented Programming Lab

Subject Code   : CSEG 2116

No. of credits : 1

Semester   : IV

Session   : January 2020

Batch   : 2018-2022

Prepared by   : Ravi Tomar

Email   : rtomar@ddn.upes.ac.in

Approved by HOD

UPES Campus   Tel : +91-135-2770137

"Energy Acres"   Fax : +91 135- 27760904

P.O. Bidholi, , Dehradun

A. PRE-REQUISITES

- Knowledge of Object Oriented programming using C++

B. PROGRAM OUTCOMES (POs) for B.Tech. in Computer Science and Engineering with Specialization in Mobile Computing

After completion of the program the student will be able to:

PO1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7:Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9: Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12:Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES

PSO1: Perform system and application programming using computer system concepts, concepts of Data Structures, algorithm development, problem solving and optimizing techniques.

PSO2: Apply software development and project management methodologies using concepts of front-end and back-end development and emerging technologies and platforms.

PSO3: ------------------to be made for each vertical----------------------
.

    1=weakly mapped
    2= moderately mapped
    3=strongly mapped

C. OBJECTIVES OF COURSE

- The student should be able to design and code the programs using java concept.
- The student should be able to understand the flexibility and modularity provided by OOPs using Java.

D. COURSE OUTCOMES (COs)

Upon completion of this course the learners will be able to:

CO1	Memorize basic concepts of object-oriented programming.
CO2	Demonstrate real world object-oriented concepts and incorporate into the Java programming language.
CO3	Execute programs, which are based on the concept of packages, interfaces and Exceptions.
CO4	Implement the concepts of Strings, Threads and collections.
CO5	Develop Java programs based on JDBC, JSP and servlets.

Table: Correlation of POs v/s COs

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	1	1	1		1								2		
CO2	2	2	2		1								2		
CO3	2	3	2		2								2		
CO4	2	3	2		2								2		
CO5	2	3	3		2								2		

1=weakly mapped
2= moderately mapped
3=strongly mapped

Course Code	Course Title	Engineering Knowledge PO1	Problem analysis PO2	Design/development of solutions PO3	Conduct investigations of complex problems PO4	Modern tool usage PO5	The engineer and society PO6	Environment and sustainability PO7	Ethics PO8	Individual or team work PO9	Communication PO10	Project management and finance PO11	Life-long Learning PO12	Perform system and application programming using computer PSO1	Apply software development and project management methodologies PSO2	PSO3
CSEG 2116	Advance Programming using Java	1.8	2.4	2.0	0.0	1.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	2.0	0.0	0.0

Mapping between COs and POs		
CO1	Memorize basic concepts of object oriented programming.	PO1 PO2 PO3 PO5 PSO1
CO2	Demonstrate real world object-oriented concepts and incorporate into the Java programming language.	PO1 PO2 PO3 PO5 PSO1
CO3	Execute programs, which are based on the concept of packages, interfaces and Exceptions.	PO1 PO2 PO3 PO5 PSO1
CO4	Implement the concepts of Strings, Threads and collections.	PO1 PO2 PO3 PO5 PSO1
CO5	Develop Java programs based on JDBC, JSP and servlets.	PO1 PO2 PO3 PO5 PSO1

E. COURSE OUTLINE

Expt. No.	Big Ideas/ Topics	Modality
1	Introduction to Java Programming-Installation Process	F2F
2,3	Basic Java Programming	F2F
4	Inheritance	F2F
5	Interfaces	F2F
6	Packages	F2F
7	Exceptions	F2F
8	String Handling and Wrapper Class	F2F
9	Threads and Collections	F2F
10	JDBC	F2F
11	Servlets	F2F
12	JSP	F2F

Expt. No.	Big Ideas/ Topics	COs
1	Introduction to Java Programming-Installation Process	CO1
2,3	Basic Java Programming	CO1, CO2
4	Inheritance	CO2
5	Interfaces	CO3
6	Packages	CO3
7	Exceptions	CO3
8	String Handling and Wrapper Class	CO4
9	Threads and Collections	CO4
10	JDBC	CO5
11	Servlets	CO5
12	JSP	CO5

F. PEDAGOGY

In continuation to problem description, the solution to the given problem statement should be designed suitably using algorithm/flow-chart/pseudocode. After obtaining a successful design, the design is implemented using java language and tested with appropriate test cases (with an insight on Input/Output Data Constraints). Students are evaluated under two main categories (1) Performance (via efficient design and implementation) and record, and (2) Preparation of the student evaluated via viva-voce /quiz. The same is detailed in Section-E.

G. COURSE COMPLETION PLAN

No. of Experiments planned/taken		No. of Quizzes/ viva planned/ conducted	
12	-	3	-

One lab session = 120 min.

H. EVALUATION & GRADING

Students will be evaluated regularly/continuously throughout the course based on the following:
1) Performance & Record   -   50%
2) Viva Voce or Quiz Examination   -   50%

H.1 Performance & Record: WEIGHTAGE - 50%

11 lab Experiments are conducted face-to-face (F2F). The lists of activities performed under the experiments are detailed in Section-F. A sample template of the evaluation of lab activity is provided in the table below.

Evaluation of each Lab Activity	
Problem Description	10% marks
Algorithm Design	40% marks
Coding Syntax, Execution and Bug Fixing	30% marks
I/O Test cases & Data Constraints	10% marks
Records (submitted before the very next turn.)	10% marks

H.2 Viva Voce or Quiz Examination: WEIGHTAGE - 50%

The preparation of the students would be evaluated based on two viva-voce or quiz examinations in periodic schedules (each with 50% weightage).

It is mandatory for the students to attend the above said continuous evaluation. Students who do not attend will lose their marks. Continuous Internal Assessment Record Sheet will be displayed at the end of the semester.

GRADING:

The overall marks obtained at the end of the semester comprising the above two mentioned shall be converted to a grade.

Student(s), who have met the qualifying criteria of individual practical subject but not met qualifying criteria of SGPA, will not be allowed to re-appear for improvement. However those students with Grade "F" and those who wish to re-appear in the practical subject, shall be required to pay the prescribed fee per subject as notified by the University. These students will be eligible to *repeat continuous evaluation* of that respective practical subject(s) during summer vacation (June-July).

Grade shall be awarded on the performance of the student(s). The Grade will be capped as per the rules mentioned in student Bulletin. All Other rules and regulations such as requirement of passing, etc. will remain same as mentioned in rules & regulations.

I. <u>DETAILED DELIVERY PLAN</u>

<u>EXPERIMENT – 1</u>

TITLE: Introduction to Java Environment

1) Installation of JDK
2) Setting of path and classpath
3) Introduction to Eclipse

<u>EXPERIMENT – 2</u>

TITLE: Basic Java Programming

1) Write a program to find the largest of 3 numbers.
2) Write a program to implement a command line calculator.
3) Write a program using classes and object in java.
4) Write a program to accept 10 student's marks in an array, arrange it into ascending order, convert into the following grades and print marks and grades in the tabular form.

Between 40 and 50 : PASS

Between 51 and 75 : MERIT

and above : DISTINCTION

EXPERIMENT – 3

1) Write a program to accept three digits (i.e. 0 - 9) and print all its possible combinations. (For example if the three digits are 1, 2, 3 than all possible combinations are : 123, 132,

   213, 231, 312, 321.)

2) Write a Java Program to accept 10 numbers in an array and compute the square of each number. Print the sum of these numbers.

3) Write a program to input a number of a month (1 - 12) and print its equivalent name of the month.(e.g 1 to Jan, 2 to Feb. 12 to Dec.)

4) Write a program to find the sum of all integers greater than 40 and less than 250 that are divisible by 5.

EXPERIMENT – 4

TITLE Inheritance

1) Write a Java program to show that private member of a super class cannot be accessed from derived classes.

2) Write a program in Java to create a Player class. Inherit the classes Cricket _Player, Football _Player and Hockey_ Player from Player class.

3) Write a class Worker and derive classes DailyWorker and SalariedWorker from it. Every worker has a name and a salary rate. Write method ComPay (int hours) to compute the week pay of every worker. A Daily Worker is paid on the basis of the number of days he/she works. The Salaried Worker gets paid the wage for 40 hours a week no matter what the actual hours are. Test this program to calculate the pay of workers. You are expected to use the concept of polymorphism to write this program.

4) Consider the trunk calls of a telephone exchange. A trunk call can be ordinary, urgent or lightning. The charges depend on the duration and the type of the call. Write a program using the concept of polymorphism in Java to calculate the charges.

5) Design a class employee of an organization. An employee has a name, empid, and salary. Write the default constructor, a constructor with parameters (name, empid, and salary) and methods to return name and salary. Also write a method *increaseSalary* that raises the employee's salary by a certain user specified percentage. Derive a subclass Manager from employee. Add an instance variable named department to the manager class. Supply a test program that uses theses classes and methods.

EXPERIMENT – 5

TITLE: Interface

1) Write a program to create interface named test. In this interface the member function is square. Implement this interface in arithmetic class. Create one new class called ToTestInt. In this class use the object of arithmetic class.

2) Write a program to create interface A, in this interface we have two method meth1 and meth2. Implements this interface in another class named MyClass.

3) Write a program in Java to show the usefulness of Interfaces as a place to keep constant value of the program

4) Write a program to create an Interface having two methods division and modules. Create a class, which overrides these methods.

EXPERIMENT – 6

TITLE: Package

1) Write a Java program to implement the concept of importing classes from user defined package and created packages.

2) Write a program to make a package Balance. This has an Account class with Display_Balance method. Import Balance package in another program to access Display_Balance method of Account class.

EXPERIMENT – 7

TITLE: Exceptions

1) Write a program in Java to display the names and roll numbers of students. Initialize respective array variables for 10 students. Handle ArrayIndexOutOfBoundsExeption, so that any such problem doesn't cause illegal termination of program.

2) Write a Java program to enable the user to handle any chance of divide by zero exception.

3) Create an exception class, which throws an exception if operand is nonnumeric in calculating modules. (Use command line arguments).

4) On a single track two vehicles are running. As vehicles are going in same direction there is no problem. If the vehicles are running in different direction there is a chance of collision. To avoid collisions write a Java program using exception handling. You are free to make necessary assumptions.

5) Write a java program to throw an exception for an employee details.
   - If an employee name is a number, a name exception must be thrown.
   - If an employee age is greater than 50, an age exception must be thrown.
   - Or else an object must be created for the entered employee details

EXPERIMENT – 8

TITLE: Strings Handling and Wrapper Class

1) Write a program for searching strings for the first occurrence of a character or substring and for the last occurrence of a character or substring.

2) Write a program that converts all characters of a string in capital letters. (Use StringBuffer to store a string). Don't use inbuilt function.

3) Write a program in Java to read a statement from console, convert it into upper case and again print on console. (Don't use inbuilt function)

4) Write a program in Java to create a String object. Initialize this object with your name. Find the length of your name using the appropriate String method. Find whether the character 'a' is in your name or not; if yes find the number of times 'a' appears in your name. Print locations of occurrences of 'a' .Try the same for different String objects

TITLE: Wrapper Classes

1) Write a Java code that converts int to Integer, converts Integer to String, converts String to int, converts int to String, converts String to Integer converts Integer to int.

2) Write a Java code that converts float to Float converts Float to String converts String to float converts float to String converts String to Float converts Float to float.

EXPERIMENT – 9

TITLE: Threads and Collections

1) Write a program to implement the concept of threading by extending Thread Class and Runnable interface.
2) Write a program for generating 2 threads, one for printing even numbers and the other for printing odd numbers.
3) Write a program to launch 10 threads. Each thread increments a counter variable. Run the program with synchronization.
4) Write a Java program to create five threads with different priorities. Send two threads of the highest priority to sleep state. Check the aliveness of the threads and mark which thread is long lasting

TITLE: Collections

1) Write a program for the following:

Read all elements from ArrayList by using Iterator.

Create duplicate object of an ArrayList instance.

Reverse ArrayList content.

2) Write a program for the following HashMap

find whether specified key exists or not.

find whether specified value exists or not

get all keys from the given HashMap

get all key-value pair as Entry objects

3) Write a program for the following HashSet

copy another collection object to HashSet object.

delete all entries at one call from HashSet

search user defined objects from HashSet

EXPERIMENT – 10

TITLE: JDBC

1) Create a database table to store the records of employee in a company. Use getConnection function to connect the database. The statement object uses executeUpdate function to create a table.

2) Create a database of employee of company in mysql and then use java program to access the database for inserting information of employees in database. The SQL statement can be used to view the details of the data of employees in the database.

EXPERIMENT – 11

TITLE: Servlet

1) Servlet: a) ServletContext interface b)getParameterValues( ) of Servlet Request

2) Write a Servlet page to display current date of the server.

3) Write a Servlet page to which include the two other Servlet page through of include directives feature provided in Servlet.

4) Write a Servlet page to create a simple calculator.

EXPERIMENT – 12

Title: JSP

Project Specification: (Write the following classes in same project named as Proj_Lab9)

Objective: After these lab exercises students will be in position to clear the concept of JSP and how to write the server side scripting language.

1)   Write a JSP page to access the data of a student from the student table.

2)   Write a JSP Login page to enter the username and password entered by

   user and display the welcome page on successful login otherwise display

   wrong authentication page.

J. SUGGESTED READINGS:
1. The Java Programming Language 3rd Edition, Ken Arnold, James Gosling, Pearson
2. A premier guide to SCJP 3rd Edition, Khalid Mughal, Pearson
3. Thinking in Java, 3rd Edition, Bruce Ackel, Pearson
4. Video resources http://www.youtube.com and blackboard.

K. GUIDELINES

Cell Phones and other Electronic Communication Devices: Cell phones and other electronic communication devices (such as Blackberries) MUST be turned off during the lab session.

E-Mail and online learning tool: Each student in the class should have UPES e-mail id and a password to access the Blackboard regularly. The best way to arrange meetings with faculty or is by email and prior appointment. Various research papers/reference material will be mailed/uploaded on online learning platform time to time.

Attendance: Students are required to have **minimum attendance of 75%** in the subject.

Passing criterion: Student has to secure minimum 40% marks of the "highest marks scored by the student for the subject" in the total marks in order to pass in that paper.

L. COURSE OUTCOME ASSESSMENT

To assess the fulfilment of course outcomes two different approaches have been decided. Degree of fulfillment of course outcomes will be assessed in different ways through direct assessment and indirect assessment. In Direct Assessment, it is measured through Continuous assessments. Each assessment is designed in such a way that it can address one or two outcomes (depending upon the course completion). Indirect assessment is done through the student survey which needs to be designed by the faculty (sample format is given below) and it shall be conducted towards the end of course completion. The evaluation of the achievement of the Course Outcomes shall be done by analyzing the inputs received through Direct and Indirect Assessments and then corrective actions suggested for further improvement.

Sample format for Indirect Assessment of Course outcomes

NAME:

ENROLLMENT NO:
SAP ID:
COURSE:
PROGRAM:

Please rate the following aspects of course outcomes of Advance Programming using Java Lab .
Use the scale 1-4* :

COs		1	2	3	4
CO1	Memorize basic concepts of object oriented programming.				
CO2	Demonstrate real world object-oriented concepts and incorporate into the Java programming language.				
CO3	Execute programs, which are based on the concept of packages, interfaces and Exceptions.				
CO4	Implement the concepts of Strings, Threads and collections.				
CO5	Develop Java programs based on JDBC, JSP and servlets.				

*

1	Below Average		3	Good
2	Average		4	Very Good

Signature