Pandit Deendayal Energy University

School of Technology

Department of Computer Science and Engineering

Odd Semester 2022-2023

Course student handout file

INDEX

Name of the course: Object Oriented Programming with Java	Course Code: 20CP204T
Program: B. Tech. Branch: CE	Semester: 3rd Academic Year: 2022-23
Name of Course Coordinator: Dr. Debabrata Swain	
Subject Teachers (Division wise/Batch wise): Div-6 1. Dr. Shivangi Surati	

Date:

Signature of Subject Teachers **Signature of Department Coordinator (IQAC)** **Signature of Head of the Department**

Departmental Vision & Mission

Vision

“To contribute to the society by imparting transformative education and producing globally competent professionals having multidisciplinary skills and core values to do futuristic research & innovations.”

Mission

- To accord high quality education in the continually evolving domain of Computer Engineering by offering state-of-the-art undergraduate, postgraduate, doctoral programmes.
- To address the problems of societal importance by contributing through the talent we nurture and research we do:
- To collaborate with industry and academia around the world to strengthen the education and multidisciplinary research ecosystem.
- To develop human talent to its fullest extent so that intellectually competent and imaginatively exceptional leaders can emerge in a range of computer professions.

Program Educational Objectives (PEOs) of Department

The Program Educational Objectives of B. Tech. (Computer Engineering) program are:

1. To prepare graduates who will be successful professionals in industry, government, academia, research, entrepreneurial pursuit and consulting firms
2. To prepare graduates who will make technical contribution to the design, development and production of computing systems
3. To prepare graduates who will get engage in lifelong learning with leadership qualities, professional ethics and soft skills to fulfill their goals
4. To prepare graduates who will adapt state of the art development in the field of computer engineering

Program Outcomes (POs)
Undergraduate engineering program are designed to prepare graduates to attain the following program outcomes:

1. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. 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.
3. 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.
4. 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.
5. 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.
6. 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.
7. 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.
8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. 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.
11. 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.
12. 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 (PSOs)

The graduates of CSE department will be able to:

1. Develop computer engineering solutions for specific needs in different domains applying the knowledge in the areas of programming, algorithms, hardware-interface, system software, computer graphics, web design, networking and advanced computing.
2. Analyze and test computer software designed for diverse needs.
3. Pursue higher education, entrepreneurial ventures and research.

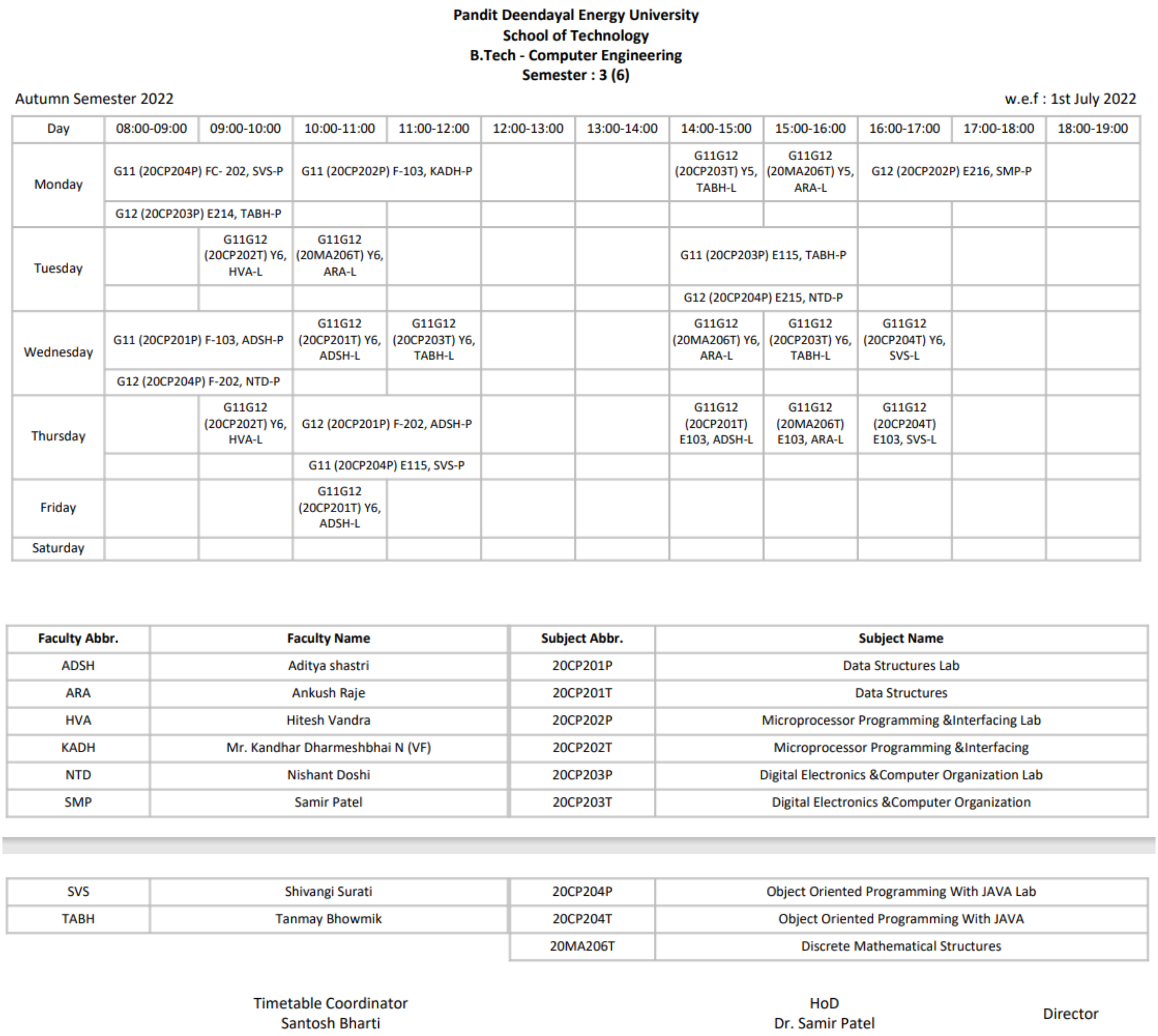
Academic Calendar

		11	12	13	14	15	16	17	FACULTY DEVELOPMENT PROGRAMME WEEK
		18	19	20	21	22	23	24	FACULTY DEVELOPMENT PROGRAMME WEEK
1	JULY 2022	25	26	27	28	29	30	31	COMMENCEMENT OF ODD SEMESTER: July 25
2	AUGUST	1	2	3	4	5	6	7	
3		8	9	10	11	12	13	14	RAKSHA BANDHAN
4		15	16	17	18	19	20	21	INDEPENDENCE DAY, JANMASHTAMI
5		22	23	24	25	26	27	28	
6	SEP	29	30	31	1	2	3	4	SAMVATSSARI
7		5	6	7	8	9	10	11	
8		12	13	14	15	16	17	18	**MID-SEM EXAMINATIONS**
9		19	20	21	22	23	24	25	
10	OCT	26	27	28	29	30	1	2	COURSE FEEDBACK WEEK
11		3	4	5	6	7	8	9	DUSSHERA
12		10	11	12	13	14	15	16	
13		17	18	19	20	21	22	23	
14		24	25	26	27	28	29	30	DIWALI WEEK
15	NOV	31	1	2	3	4	5	6	
16		7	8	9	10	11	12	13	GURU NANAK JAYANTI
17		14	15	16	17	18	19	20	COMPLETION OF ODD SEMESTER: Nov. 18
		21	22	23	24	25	26	27	**FOET Practical Exams : Nov.21 Onwards** **FOLS Sem. End Examination : Nov. 21 Onwards**
	DEC	28	29	30	1	2	3	4	**FOET Sem. End Examination : Nov.28 Onwards**
		5	6	7	8	9	10	11	
		12	13	14	15	16	17	18	Rural Internship for FOLS Students: Dec. 17, 2022 to Jan. 10, 2023
		19	20	21	22	23	24	25	
		26	27	28	29	30	31	1	**WINTER BREAK**

Total Weeks: 17

Class Time Table and Faculty Time Table with office hours

Class Time Table:



Pandit Deendayal Energy University
School of Technology
B.Tech - Computer Engineering
Semester : 3 (6)

Autumn Semester 2022

w.e.f : 1st July 2022

Day	08:00-09:00	09:00-10:00	10:00-11:00	11:00-12:00	12:00-13:00	13:00-14:00	14:00-15:00	15:00-16:00	16:00-17:00	17:00-18:00	18:00-19:00
Monday	G11 (20CP204P) FC- 202, SVS-P		G11 (20CP202P) F-103, KADH-P				G11G12 (20CP203T) Y5, TABH-L	G11G12 (20MA206T) Y5, ARA-L	G12 (20CP202P) E216, SMP-P		
	G12 (20CP203P) E214, TABH-P										
Tuesday		G11G12 (20CP202T) Y6, HVA-L	G11G12 (20MA206T) Y6, ARA-L				G11 (20CP203P) E115, TABH-P				
							G12 (20CP204P) E215, NTD-P				
Wednesday	G11 (20CP201P) F-103, ADSH-P		G11G12 (20CP201T) Y6, ADSH-L	G11G12 (20CP203T) Y6, TABH-L			G11G12 (20MA206T) Y6, ARA-L	G11G12 (20CP203T) Y6, TABH-L	G11G12 (20CP204T) Y6, SVS-L		
	G12 (20CP204P) F-202, NTD-P										
Thursday		G11G12 (20CP202T) Y6, HVA-L	G12 (20CP201P) F-202, ADSH-P				G11G12 (20CP201T) E103, ADSH-L	G11G12 (20MA206T) E103, ARA-L	G11G12 (20CP204T) E103, SVS-L		
			G11 (20CP204P) E115, SVS-P								
Friday			G11G12 (20CP201T) Y6, ADSH-L								
Saturday											

Faculty Abbr.	Faculty Name	Subject Abbr.	Subject Name
ADSH	Aditya shastri	20CP201P	Data Structures Lab
ARA	Ankush Raje	20CP201T	Data Structures
HVA	Hitesh Vandra	20CP202P	Microprocessor Programming &Interfacing Lab
KADH	Mr. Kandhar Dharmeshbhai N (VF)	20CP202T	Microprocessor Programming &Interfacing
NTD	Nishant Doshi	20CP203P	Digital Electronics &Computer Organization Lab
SMP	Samir Patel	20CP203T	Digital Electronics &Computer Organization
SVS	Shivangi Surati	20CP204P	Object Oriented Programming With JAVA Lab
TABH	Tanmay Bhowmik	20CP204T	Object Oriented Programming With JAVA
		20MA206T	Discrete Mathematical Structures

Timetable Coordinator
Santosh Bharti

HoD
Dr. Samir Patel

Director

Faculty Time table:

Shivangi Surati

Academic Year: 2022-23 (Odd semester) W.E.F.: 1st July, 2022

Day	08:00-09:00	09:00-10:00	10:00 to 11:00	11:00 to 12:00	12:00 to 13:00	13:00 to 14:00	14:00 to 15:00	15:00 to 16:00	16:00-17:00	17:00-18:00
MON	G11 (20CP204P) E-215, CP(3) - P						G5 (20CP302P) F-103, CP(5) - P			
TUE				Office Hour			G6 (20CP302P) F-103, CP(5) - P		G8 (20CP302P) F-202, CP(5) - P	
WED									G11, G12 (20CP204T) F-404, CP(3)-L	
THURS	G10 (20CP204P) F-104, CP(3) - P		G11 (20CP204P) E-115, CP(3) - P						G11, G12 (20CP204T) E-103, CP(3)-L	
FRI	G10 (20CP204P) F-203, CP(3) - P									
SAT										

Syllabus

20CP204T					Object Oriented Programming with Java					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
2	0	0	2	2	25	50	25			100

COURSE OBJECTIVES

- To build an understanding of basic concepts of object-oriented programming techniques
- To develop programming skills in Java programming language
- To implement object-oriented techniques using Java language features.
- To develop software using object-oriented programming paradigms

UNIT 1 BASICS OF JAVA 7 Hrs.

Features of Object Oriented Programming and Java, Basics of Java programming, Data types, Variables, Operators, Control structures including selection, Looping, Java methods, Overloading, Math class, Arrays in Java.

UNIT 2 INHERITANCE 7 Hrs.

Basics of objects and classes in java, Constructors, Visibility modifiers, Inbuilt classes in Java, this reference; Inheritance in java, Overriding, Object class, Polymorphism, Dynamic binding, Abstract class, Interface in java, Package in java.

UNIT 3 I/O PROGRAMMING, EXCEPTION AND MULTITHREADING 6 Hrs.

Introduction to Java IO streams, Character and Binary streams, reading data from and writing data to files, Difference between error and exception, Exception handling in Java, Multithreading in Java, Thread life cycle and methods, Runnable interface, Thread synchronization

UNIT 4 EVENT HANDLING AND GUI PROGRAMMING 6 Hrs.

Event handling in Java, GUI Components and Layouts, Applet and its life cycle.

Max. 26 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1- **Describe** the basic features of Object-oriented programming and map them with the Java.
CO2- **Distinguish** Objects and Classes using Java.
CO3- **Demonstrate** Inheritance and Runtime Polymorphism
CO4- **Apply** I/O handling, exception handling for interactive problem.
CO5- **Use** the concepts of Event Handling in GUI Programming.
CO6- **Construct** object-oriented solutions for small systems involving multiple objects.

TEXT/REFERENCE BOOKS

1. Brett D. McLaughlin, Head First Object-Oriented Analysis and Design, O Reilly, 2006
2. Matt Weisfeld, The Object-Oriented Thought Process, Addison-Wesley Professional, 2019
3. Herbert Schildt, The Complete Reference, Java 2, McGraw Hill, 2020
4. Balagurusamy, Programming with Java – A Primer, McGraw Hill, 2019

END SEMESTER EXAMINATION QUESTION PAPER PATTERN

Max. Marks: 100 **Exam Duration: 3 Hrs**

Part A: 10 Questions of 2 marks each-No choice	20 Marks
Part B: 2 Questions from each unit with internal choice, each carrying 16 marks	80 Marks

Lesson Plan

Lecture No.	Topic to be covered	Teaching Aid to be used	Remarks (Text book/Unit No etc.)
1	Syllabus Discussion, Course Introduction	BW+PPT	Unit 1
2	Features of Object Oriented Programming and Java,	BW+PPT	Unit 1
3	Basics of Java programming,	BW+PPT	Unit 1
4	Data types, Variables, Operators,	BW+PPT	Unit 1
5	Control structures including selection, Looping	BW+PPT	Unit 1
6	Java methods, Overloading	BW+PPT	Unit 1
7	Math class, Arrays in java	BW+PPT	Unit 1
8	Basics of objects and classes in java	BW+PPT	Unit 2
9	Constructors, Visibility modifiers,	BW+PPT	Unit 2
10	Inbuilt classes in Java, this reference	BW+PPT	Unit 2
11	Inheritance in java, Overriding Object class	BW+PPT	Unit 2
12	Polymorphism, Dynamic binding	BW+PPT	Unit 2
13	Abstract class, Interface in Java	BW+PPT	Unit 2
14	Package in java	BW+PPT	Unit 2
15	Introduction to Java IO streams, Character and Binary streams	BW+PPT	Unit 3
16	reading data from and writing data to files	BW+PPT	Unit 3
17	Difference between error and exception, Exception handling in Java	BW+PPT	Unit 3
18	Multithreading in Java, Thread life cycle and methods	BW+PPT	Unit 3
19	Runnable interface	BW+PPT	Unit 3
20	Thread synchronization	BW+PPT	Unit 3
21	Event handling in Java - 1	BW+PPT	Unit 4
22	Event handling in Java - 2	BW+PPT	Unit 4
23	GUI Components and Layouts - 1	BW+PPT	Unit 4
24	GUI Components and Layouts - 2	BW+PPT	Unit 4
25	Applet and its life cycle - 1	BW+PPT	Unit 4
26	Applet and its life cycle - 2	BW+PPT	Unit 4

Legends: BW (Board Work), PPT (PowerPoint Slides)

Course Articulation Matrix

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PS01	PSO2	PSO3
CO1	1	1	3	1	3	1	1	1	1	-	3	1	1	2	-
CO2	1	1	3	1	2	1	1	1	1	-	3	1	1	1	-
CO3	1	1	3	1	2	1	1	1	3	-	3	1	1	1	-
CO4	1	1	3	1	2	1	1	1	3	-	3	1	1	1	-
CO5	1	2	3	1	2	1	1	1	1	-	3	1	2	1	-
CO6	1	2	3	1	2	2	1	1	1	-	3	1	2	1	-
	1	1.3	3	1	2.1	1.1	1	1	1.6	-	3	1	1.3	1.1	-

Program Articulation Matrix

PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PS01	PSO2	PSO3
1	1.3	3	1	2.1	1.1	1	1	1.6	-	3	1	1.3	1.1	-

Correlation levels 1, 2 or 3 as defined below:

1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

Evaluation Scheme and Rubrics

Course code: 20CP204T **Course name:** Object Oriented Programming with Java

Course Outcomes (COs): On completion of the course, students will be able to
CO1- Describe the basic features of Object-oriented programming and map them with the Java.
CO2- Distinguish Objects and Classes using Java.
CO3- Demonstrate Inheritance and Runtime Polymorphism
CO4- Apply I/O handling, exception handling for interactive problem.
CO5- Use the concepts of Event Handling in GUI Programming.
CO6- Construct object-oriented solutions for small systems involving multiple objects.

CO Assessment Tools (Direct Assessment):
Various assessment tools used to evaluate CO's (Rubrics) and the frequency with which the assessment processes are carried out are listed below.

Assessment Method	Assessment Tool	Description	Marks	Mapping with CO	Contribution to CO's
Direct	MID-Sem Examination	Topics to be covered till Unit - 2	50	CO1,CO2, CO3	It fractionally contributes to 25% weightage of Direct Assessment to CO attainment.
	Total 50 marks will be converted into 25 marks for the mid-sem eval.				
Direct	Assignments	For each unit a separate assignment will be prepared	25	CO1,CO2, CO3,CO4, CO5, CO6	It fractionally contributes to 25% weightage of Direct Assessment to CO attainment.
	Viva	At the end of semester Question-answer based evaluation on One-to-one basis	25	CO1,CO2, CO3,CO4, CO5, CO6	
	Total 50 marks will be converted into 25 marks for the continuous assessment.				
Direct	End-Sem Examination	**Topics to be covered:** Unit I, II, III, IV	40	CO1,CO2, CO3,CO4, CO5, CO6	It contributes to 50% weightage of Direct Assessment to CO attainment.
	Total 100 marks will be converted into 50 marks at the end.				

Tutorials, Assignments, Case Studies, Quiz, Presentations etc.

Available online on Teams Platform, PPT sent to students through Email and Teams.

Lecture

- Topic wise PPTs for All units 1 to 4 will be shared with students with following file names
  - Unit 1 – **BASICS OF JAVA**
  - Unit 2 - **INHERITANCE**
  - Unit 3 – **I/O PROGRAMMING, EXCEPTION AND MULTITHREADING**
  - Unit 4 – **EVENT HANDLING AND GUI PROGRAMMING**

LAB

- Unit wise practical list is covered and will be distributed to students with synchronization to class teaching.

Copy of Mid and End semester Examination Question Papers (Old and Current), solution of current examination with stage-wise marking scheme

Type: MCQ

Q1. See the following Java Code

```
class Fuel{  public int litre = 0;  }

class Petrol extends Fuel{

Public static void fill(){

        litre = 10;

        super.litre = 20;

        new Fuel().litre = 30;

        }

}
```

Which statement(s) can not access and modify the litre variable of the fuel class from the fill() method? *A.* litre = 10; *B.* super.litre = 20; *C.* new Fuel().litre = 30; (1)

1. C
2. A and C
3. **A and B
4. All can access.

Q2. Predict the output of following Java program.

```
class Test {
    public static void main(String[] args) {
      for(int i = 0; 0; i++)
      {
          System.out.println("Hello");
          break;
      }
    }
}
```

(1)

1. Hello
2. Empty Output
3. **Compiler Error
4. Runtime Error

Q3. Which statement is true regarding an object? (1)

1. An object is what classes instantiated are from

2. **An object is an instance of a class
3. An object is a variable
4. An object is a reference to an attribute

Q4. In object-oriented programming, new classes can be defined by extending existing classes. This is an example of: (1)

1. Encapsulation
2. Interface
3. Composition
4. **Inheritance

Q5. Which of the following statements is/are TRUE regarding JAVA ? (a) Constants that cannot be changed are declared using the ‘static’ keyword. (b) A class can only inherit one class but can implement multiple interfaces. (1)

1. Only (a)
2. **Only (b)
3. Both (a) and (b)
4. None of (a) and (b)

Q6. **Object-oriented inheritance models the: (1)**

1. **“is a kind of” relationship
2. “has a” relationship
3. “want to be” relationship
4. “contains” of relationship.

Q7. **Predict the output of following Java program**

```
class Main {
  public static void main(String args[]) {
    try {
      throw 10;
    }
    catch(int e) {
      System.out.println("Got the  Exception " + e);
    }
  }
}  (1)
```

1. Got the Exception 10
2. Got the Exception 0
3. **Compiler Error
4. System dependent Output

Q8. **Predict the output of following Java program**

```
class Base extends Exception {}
class Derived extends Base  {}

public class Main {
  public static void main(String args[]) {
   // some other stuff
   try {
       // Some monitored code
       throw new Derived();
    }
    catch(Base b)     {
       System.out.println("Caught base class exception");
    }
    catch(Derived d)  {
       System.out.println("Caught derived class exception");
    }
  }
}  (1)
```

1. Caught base class exception
2. Caught derived class exception
3. Compiler Error because derived is not throwable
4. **Compiler Error because base class exception is caught before derived class

Q9. **When a class extends Thread class, it should override _____ method of Thread class.** (1)

1. start()
2. **run()
3. init()
4. go()

Q10. System class is defined in _____ package. (1)

1. java.util
2. java.io
3. **java.lang
4. No packages

Q11. What will be the output of following Java code:

```
class Main {
 public static void main(String args[]){
   final int i;
   i = 20;
   i = 30;
   System.out.println(i);
 }
} (1)
```

1. 30
2. **Compiler Error
3. Garbage Value
4. 0

Q12. Which of these field declarations are legal within the body of an interface? (1)

1. Private final static int answer = 42
2. **public static int answer=42
3. final static answer =42
4. all are valid

Q13. A method within a class is only accessible by classes that are defined within the same package as the class of the method. Which one of the following is used to enforce such restriction? (1)

1. Declare the method with the keyword public
2. Declare the method with the keyword private
3. Declare the method with the keyword protected
4. **Do not declare the method with any accessibility modifiers
5. Declare the method with the keyword public and private.

Q14. What is the fundamental unit of information of writer streams? (1)

1. **Character
2. Byte
3. File
4. Record

Q15. Which of the following events will cause a thread to die? (1)

1. The method sleep( ) is called
2. The method wait( ) is called

3. Execution of the start( ) method ends

4. **Execution of the run( ) method ends

5. Execution of the run() method is called.

Q16. Given a class named Student, which one of these is a valid constructor declaration for the class? (1)

1. **Student(Student b) { }

2. Student Student() { }

3. private final Student() { }

4. void Student() { }

5. abstract Student() { }

Q17. Which one of the following class definitions is a valid definition of a class that cannot be instantiated? (1)

1. class Ghost

    {

    abstract void haunt();

    }

2. abstract class Ghost

    {

    void haunt();

    }

3. **abstract class Ghost

    {

    void haunt() { };

    }

4. abstract Ghost

    {

    abstract void haunt();

    }

Q18. How restrictive is the default accessibility compared to public, protected, and private accessibility? (1)

1. Less restrictive than public

2. More restrictive than public, but less restrictive than protected
3. **More restrictive than protected, but less restrictive than private
4. More restrictive than private

Q19.

```
interface Charger {
        void fill();
}

interface Fuel {
        void fill();
}

class Car implements Charger, Fuel {

        @Override
        public void fill() {
                System.out.println("filling");
        }
}
```

Choose the correct answer about the interface implementation in the code. (1)

1. Two interfaces cannot have a method with the same name.
2. Compile-time error.
3. Run-time error because of ambiguous call of the fill() method.
4. **The interface implementation is fine.

```
Q20. public class Sample {

public static void main(String[] args) {

if (new String("Java") == "Java");//1
if (new String("Java") == new String("Java"));// 2
if (new String("Java").equals("Java")); // 3

        }
}
```

The expressions in the *if* conditions 1, 2, and 3 will evaluate to: (1)

1. true, true, false
2. false, true, false
3. **false, false, true
4. true, false, true

Type: DES

Q21. Write a Java program that has one function with two arguments. One argument is the position (index value) and second argument is the value. The function places (or replaces) the value in the argument on the supplied index position and returns the original value that was replaced with this new value. (For example, let an array of 5 integers have the values 10, 15, 3, 25, 17. If we pass to the function the values (3, 90), then on the index position 3 the value of 25 will be replaced with this new value 90. The function will return 25.) (5)

Q22. Write a Java program having one function that takes a number and prints all its prime factors. (5)

Q23. Write a program that calculates the sum of integers provided thru command line arguments. It also checks if the command line argument contains any non-integer values and report such values in the output. (5)

Q24. Discuss the applications when to use Ragged (a.k.a jagged) Arrays and when to use ArrayList? (5)

Q25. With suitable examples explain the syntax and usage of abstract class and final class. (5)

Q26. With suitable example explain the use of Inner class. (5)

Q27. Write a Java program that implements a multi-thread application that has three different threads. (5)

Q28. Create a student data entry form in Java using JForm class. The form should have the following fields : Enrollment No., Name, Email ID, contact no., Admission Year. Branch. The branch field should be a drop-down list. (5)

Roll No. ___________

Pandit Deendayal Energy University

(Formerly Pandit Deendayal Petroleum University)

End-Semester Re-Examination

B.Tech. (Information and Communication Technology)

Semester - III

Date: 23/05/2022
Time: 2hrs
Max. Marks: 40

Course Name: Object Oriented Concepts and Programming
Course Code : 20IC203T

Instructions:

1. Do not write anything other than your roll number on the question paper.
2. Assume suitable data wherever essential and mention it clearly.
3. Writing appropriate units, nomenclature, and drawing neat sketches/schematics wherever required is an integral part of the answer.

Q – 1	Discuss the difference between Object-oriented and procedure-oriented programming approaches with a case study	[10]	CO1
Q - 2	(a) With the figure, explain the difference between multiple and multi-level Inheritance	[05]	CO2
	(b) Let a software for Library has to be created using Object-oriented approach. Describe at least three classes that you will create to design the software. Also, specify the relationship/ association between those classes.	[05]	CO3
Q - 3	(a) From the following Java code snippets identify the errors. (2 marks to each)	[04]	CO4

```
class Main{
public static void main(String[] args)
{
  int a=0,b =5,c=6;
 if(a>b)
  {
    if(a>c){
      for(i = 0;i<5;i++)
       System.out.println(i);
  }
}}
```

```
class Main{
public static void main(string[] args)
{
 int a=0,b =5,c=6;
 c = a+b;
System.out.print(c+);
}


}
```

	(b) Provide the Output for following code snippets written in Java (3 marks to each):	[06]	CO5

```
class Main{
  public static void main(String
args[]){
 A obj1 = new A();
A obj2 = new A();
System.out.println(obj1.x);
System.out.println(obj2.x);
 } }
Class A
{  static int x=0;
   A() {x++;}  }
```

```
class Main {
  public static void main(String arg[])
  {
        int arr[][]={{4,3},{2,1}};
        int i,j;
       for(i=1;i>-1;i--)
           {   for(j=1;j>-1;j--)
                        System.out.print(arr[i][j]);
           }
} }
```

Q – 4	Write a Java program to create a class Stone, which has an instance variable mass and a static variable total_mass. Both are of type integer. The Stone class will have a constructor that takes as an argument an integer value and assigns that value to the instance variable mass. In the main method create the 5 objects of Stone class. For each object get an integer value between 1 to 10 from user and pass it as an argument to the constructor. The total_mass is the summation of all mass quantities of all stone objects.	[10]	CO6
	OR		
Q – 4	Write a Java program to create an interface Vehicle. The Vehicle interface will have two methods, “setSpeed()” and “getSpeed()”. The purpose of “setSpeed()” method is to set the speed of a vehicle. The “getspeed()” method displays the speed of vehicle. Create two classes “Two-wheeler” and “Fourwheeler”, that will implement the Vehicle interface. Both the classes will have a variable *speed*. For two-wheeler the speed cannot be increased beyond the 60, and for Four-wheeler, the speed can not be set beyond 80 Kms. With these rules, do the proper implementation and create the objects in main method.	[10]	CO6

Roll No. ___________

Pandit Deendayal Petroleum University

Take Home Assignment
B. Tech.(CSE/ICT.)
Semester - IV

Date:15/06/2020

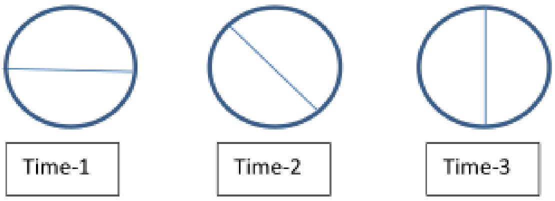
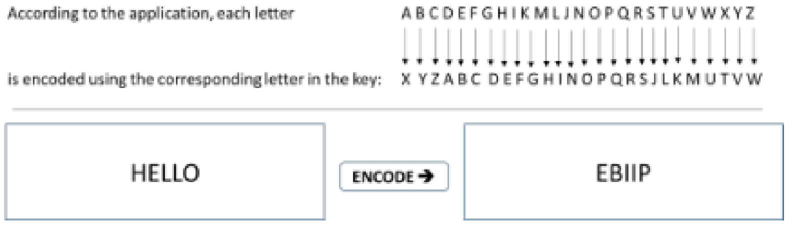
Course Name : Object Oriented Concepts and Programming
Course Code : 18CP218T **Max. Marks:50**

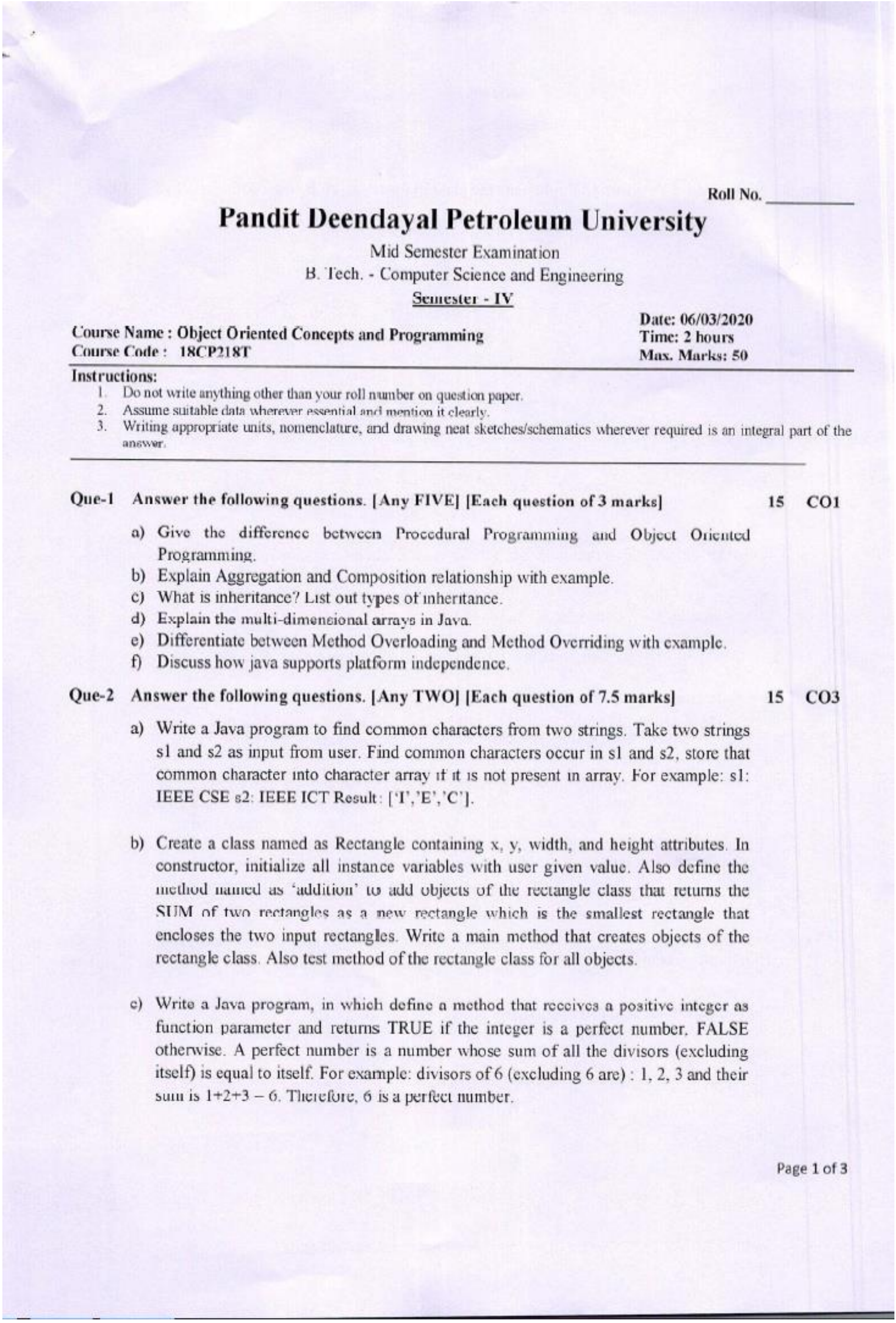
Instructions:

1. Submit legible hand written assignment on foolscap A4 size pages as single pdf- named with your roll no. and name.
2. Write your name, roll no., subject name and code at top of the assignment.
3. Assume suitable data wherever essential and mention it clearly.
4. Writing appropriate units, nomenclature, and drawing neat sketches/schematics wherever required is an integral part of the answer.
5. Submit assignment online through TCSiON or LMS (Moodle) only.

Part-A
ANSWER ALL QUESTIONS (5 x 4 Marks = 20 Marks)

Question No.	Description	Marks	Course Outcome (CO)
Que-1	Other than Polymorphism, what are the applications of Interface in software development? Explain with the use of interfaces defined in Collection framework.	04	CO4
Que-2	If I don't put my class in any package, what can be the consequences?	04	CO5
Que-3	Adapter classes are an alternative of interface implementation. In which conditions, we should prefer use of MouseAdapter instead of Mouselistener? How is it possible to use both MouseAdapter and Applet class to create an applet?	04	CO3
Que-4	Create swing based GUI application having three text areas. The first text area allows you to input a passage of text. The second text area displays a frequency analysis of the letters used in the paragraph when user click on button to analyze the text. By clicking on analyze text button, the third text area also displays frequency analysis of the word used in the paragraph. Add buttons to analyze the text and to clear the three text areas.	04	CO5
Que-5	Create a java program for "Credit Card Number Validation". Define at least two to three rules to validate credit card number, and implement it through exception handling mechanism. For example, one rule for credit card validation can be "Number must be 16 digits". GUI is not required for this application.	04	CO4
	Part-B **ANSWER ALL QUESTIONS (3 x 10 Marks = 30 Marks)**		
Que-1	Create a multi-threading program for elevator operation. There is one elevator class with two static variables Status and count. The status is a boolean variable which states whether the elevator is open or closed. The count is a variable that specifies the number of persons currently in elevator. There is one class rider, which has the method Use_Elevator. The Use_Elevator method will take as an argument the elevator object. If the	10	CO6

	elevator's current status is open then rider can use the elevator for 3 seconds (It is assumed that after 3 seconds the rider will be out from the elevator). At that time the count will be increased by one. If the count reaches to threshold value then the status of elevator will be "Closed". Once the rider will stop using elevator (after 3 seconds), then the count will be decreased by one. Implement thread concept to share the elevator between multiple riders. The programmer can assume rest of the details and create other variables, methods and classes if required.		
Que-2	Create an applet that has a circle with a single rod which passes from the center. Using thread create an animation such that it looks like the circle is moving from one end of applet to other end. (The circle should continuously change its position in horizontal direction. The rod should change its direction with changes in x,y coordinates).  	10	CO6
Que-3	Create "Encode Text" GUI based Java application as per the below given diagram. Each letter is encoded using corresponding letter in the key, when user click on "Encode" button. Each time key should be generated randomly.  According to the application, each letter is encoded using the corresponding letter in the key: A B C D E F G H I K M L J N O P Q R S T U V W X Y Z X Y Z A B C D E F G H I N O P Q R S J L K M U T V W 	10	CO5



Roll No. __________

Pandit Deendayal Petroleum University

Mid Semester Examination

B. Tech. - Computer Science and Engineering

Semester - IV

Course Name : Object Oriented Concepts and Programming
Course Code : 18CP218T

Date: 06/03/2020
Time: 2 hours
Max. Marks: 50

Instructions:

1. Do not write anything other than your roll number on question paper.
2. Assume suitable data wherever essential and mention it clearly.
3. Writing appropriate units, nomenclature, and drawing neat sketches/schematics wherever required is an integral part of the answer.

Que-1 Answer the following questions. [Any FIVE] [Each question of 3 marks] 15 CO1

a) Give the difference between Procedural Programming and Object Oriented Programming.

b) Explain Aggregation and Composition relationship with example.

c) What is inheritance? List out types of inheritance.

d) Explain the multi-dimensional arrays in Java.

e) Differentiate between Method Overloading and Method Overriding with example.

f) Discuss how java supports platform independence.

Que-2 Answer the following questions. [Any TWO] [Each question of 7.5 marks] 15 CO3

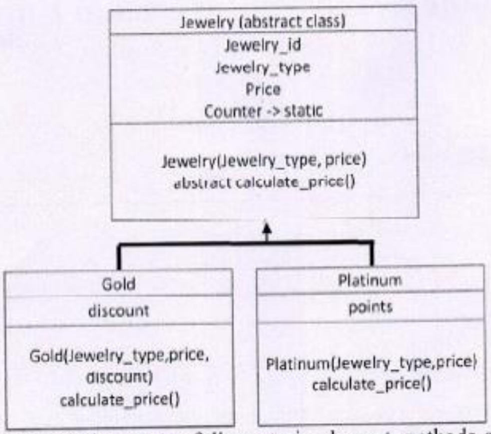
a) Write a Java program to find common characters from two strings. Take two strings s1 and s2 as input from user. Find common characters occur in s1 and s2, store that common character into character array if it is not present in array. For example: s1: IEEE CSE s2: IEEE ICT Result: ['I','E','C'].

b) Create a class named as Rectangle containing x, y, width, and height attributes. In constructor, initialize all instance variables with user given value. Also define the method named as 'addition' to add objects of the rectangle class that returns the SUM of two rectangles as a new rectangle which is the smallest rectangle that encloses the two input rectangles. Write a main method that creates objects of the rectangle class. Also test method of the rectangle class for all objects.

c) Write a Java program, in which define a method that receives a positive integer as function parameter and returns TRUE if the integer is a perfect number, FALSE otherwise. A perfect number is a number whose sum of all the divisors (excluding itself) is equal to itself. For example: divisors of 6 (excluding 6 are) : 1, 2, 3 and their sum is 1+2+3 – 6. Therefore, 6 is a perfect number.

k

Que-3 **Write a Java Program to implement the class diagram given below.** **10** **CO2**



Consider the class description as per follows to implement methods given in the class diagram.

- **Jewelry Class**
  - Initialize static variable counter to 100
  - In the constructor, auto-generate string Jewelry_id starting from 101 prefixed by "G" for Gold and "P" for Platinum class. Example – G101, P102, P103, G104 etc.
- **Gold Class**
  - calculate_price() : Add 5% service tax and discount on the price of the Jewelry class and update attribute price with new value.
- **Platinum Class**
  - calculate_price(): Add 10% service tax on the price of the Jewelry class and update attribute price with new value. Also update points based on rules as per follows: Platinum Jewelry with price more than Rs. 90000, earns 20 points and anything less than or equal to that earn 7 points.

Invoke Parent class constructor whenever it is require. Write a main method that creates object of the Gold and the Platinum class. Also test calculate_price() method for all objects.

Que-4 **Find errors from following programs if any, otherwise write output of the program. [Each question of 2.5 marks]** **10** **CO2, CO3**

a)
```
public class P1 {
public static void main(String[] args) {
int[] sample = {1,2,3,4,5};
int[] test = new int[5];
System.arraycopy(sample, 0, test, 0, 5);
test[2]=101;
System out println(test[2]+sample[2]);
} }
```

b)
```
public class P4 {
  int x = 10;
public static void main(String
args[]) {
 P4 obj;
 System.out.println(obj.x);} }
```

```
c)  class child extends Test{
        child(String s1) {
                this.s1+=s1;
                System.out.println(this.s1);
} }
public class Test {
        String s1="Famous Painting";
        Test(){s1+=" By Leonardo da
Vinci"; }
        public static void main(String[]
args) { child c1= new child("Mona Lisa");
}}
```

```
d)  class P2{
public static void main(String
args[]){
  int a=4;
  int b=8;
  if((b-a--)>=a) {
    if((b+a)%2==1)
System.out.println(a*b);
  else
System.out.println(b+a);}
  else
 System.out.println(b/a); }}
```

Course covered beyond syllabus

An introduction to Java related technologies trending in current industry will be given to students through some expert lectures conducted by Industry persons.

Materials from national and international level like NPTEL, Web resources, etc. is shared related to subject domain

- Database Connectivity using JDBC
- Socket Programming

Students will be motivated and guided to get the knowledge of latest updates in Java language thru following websites:

- www.javapoint.com
- www.tutorialspoint.com/java
- https://www.w3schools.com/java/
- https://docs.oracle.com/javase/tutorial/