Department of Computer Science and Information Systems
BITS Pilani
Object Oriented Programming (CSF213)
Comprehensive Examination *(Completely closed book)*

Marks: 120 Points

Note: ------- READ THIS---

1. **Attempt all the parts of question together. Any part attempted separately won't be checked.**
2. **Give succinct and complete answers.**
3. **In case your answer contains multiple points/ reasons: Mention all the points first before explaining any point. Underline key points.**

---

1. What does it mean for the programmers when we use the following terms?

a. Program to an interface and not to implementation.
b. Encapsulate what varies.
c. Favor composition over inheritance
d. Classes should be open for extension and closed for modification.
e. Strive for loosely coupled designs between objects that interact.

Discuss their meaning and how they are applied in Java programming?
What is/are the benefit of using the above strategies/ guiding principles in application life-cycle?
Give an example for each with UML diagram with an explanation or Java code with explanation)
Note: Answer each point separately **[(3+2+3)X 5 = 40 Marks]**

2. We are implementing a design pattern that restricts the instantiation of a class to one object.

```
class Singleton
{
    private static Singleton obj;

    // private constructor to force use of
    // getInstance() to create Singleton object
    private Singleton() {}

    public static Singleton getInstance()
    {
        if (obj==null)
            obj = new Singleton();
        return obj;
    }
}
```

[10 Marks]

What can be two big issues in the above code. Discuss their implications and share the updated java program to remove these issues.

3. Critically reason how State design pattern is different from Memento design pattern?
    a) Explain the purpose of each of them.
    b) Share the example, with UML where they are useful
    c) Write the main code implementing the UML diagrams of state and Memento design patterns. **[5+5+5 +5 = 20 Marks]**

4. Given below is the code for the producer/ consumer problem. It consists of four classes: **Q**, the queue that you're needs to be synchronized**; Producer**, the threaded object that is producing queue entries; **Consumer**, the threaded object that is consuming queue entries; and **PC**, the class that creates the single Q, Producer, and Consumer.

```
class Q {
int n;
synchronized int get() {
System.out.println("Got: " + n);
return n;
}
synchronized void put(int n) {
this.n = n;
System.out.println("Put: " + n);
}
}
```

```
class Producer implements Runnable {
Q q;
Producer(Q q) {
this.q = q;
new Thread(this, "Producer").start();
}
public void run() {
int i = 0;
while(true) {
q.put(i++);
}
}
}
```

```
1  class Consumer implements Runnable {
2   Q q;
3  Consumer(Q q) {
4   this.q = q;
5   new Thread(this, "Consumer").start();
6   }
7  public void run() {
8  while(true) {
9   q.get();
10  }
11  }
12  }
```

```
1  class PC {
2  public static void main(String args[]) {
3   Q q = new Q();
4   new Producer(q);
5   new Consumer(q);
6   System.out.println("Press Control-C to stop.");
7   }
8   }
```

We expect the following output
Put: 1
Got: 1
Put: 2
Got: 2
Put: 3
Got: 3
Put: 4
Got: 4
Put: 5
Got: 5

However, we don't get the expected output from the above program.

a) Why does this program not give the expected output?
b) Spot the changes that you shall make to get what you want. [just write the class and line number(s) which you change/delete / or add].
c) Why do your proposed changes get the desired output? **[10 marks]**

5. The following (incomplete) code reads input from file using scanner.

```
1  File file = new File("C:\\Users\\OOP\\Desktop\\test.txt");
2      Scanner sc = new Scanner(file);
3
4      while (sc.hasNextLine())
5        System.out.println(sc.nextLine());
6    }
```

We know that Java can take input in two ways: - byte streams or Character streams.

a) How is Scanner internally executed. What are the packages/classes/methods does it use?

b) Demystify the working of scanner and explain as if you are explaining it to one of your peers. **[10 marks]**

6. You tell one of your juniors that Generics improve type safety. He does not agree with you and argues that "Object references also create generic code. What is the need of using generics explicitly?"

   In order to prove your point, share the code that declares more than one type parameter in the generic type and uses it to avoid run time exceptions. Give the algorithm of what you are writing and write the complete code. **[10 marks]**

7. Give a succinct and complete understanding of following keywords in Java.

   [Don't use more than 4 lines/ 70 words to explain each. Not following the word limit will attract penalty]

   i. throw & throws
   ii. try, catch and finally
   iii. static
   iv. final
   v. synchronized
   vi. collection class
   vii. HashSet & ArrayList
   viii. Comparable & Comparator
   ix. Shallow copy and deep copy of a class
   x. BufferedOutputStream

   [2X10 = 20 marks]