University of Dhaka
Department of Computer Science and Engineering
2nd Year, 1st Semester Final Examination, 2021
CSE-2102: Object Oriented Programming

Full Mark: 70 Time: 3 Hours

Answer any 5 (five) from the following questions

1. (a) What do you understand about platform independence of a programming language? How does Java achieve this? 4

(b) Briefly explain the following concepts of Object-Oriented Programming (OOP): abstraction. encapsulation, inheritance, and polymorphism. 4

(c) The following is a typical implementation of stack data structure written in procedural language (C in this case): 6

```
void push (int st[], int tos, int num){
     st[tos] = num;
}
int pop(int st[], int tos){
     return st[tos];
}
main(){
   int stack[5];
   int tos=0;
   push(stack, tos, 45);
   tos++;
   push(stack, tos, 4);
   tos++;
   tos--;
   int item = pop(stack, tos);
   printf("%d", item);
   tos--;
   int item = pop(stack, tos);
   printf("%d", item);
}
```

Rewrite the above code in Java so that the OOP principles are enforced as much as possible.

2. (a) Write down the differences between the four access modes of Java: default case. private, protected. and public. (You do not need to give examples, just answer in text.) 4

(b) Mention three cases where the `final` keyword is used and why. 4

(c) When is a static block of code executed? Write down the output of the following code segment: 2+4

```
class A {
     static int a;
     static int b;

     static {
         System.out.println("a: " + a + ", b: " + b);
     }
```

```
        {
            b++;
            System.out.println("b in block: " + b);
        }
        public A(){
            System.out.println("constructor");
        }
        public static void main(String[] args) {
            System.out.println("Some text.");
            A ob = new A();
            A ob2 = new A();
        }
    }
```

3 (a) Consider the following code. Rewrite the code using inheritance and the keyword super. You must write optimal code in terms of OOP principles. You should not use more classes than A, B, C and Demo. 5

```
class C{
  int a, b, c;
  C(int a, int b, int c){
     this.a=a; this.b=b; this.c=c;
  }
}
class B {
  int a, b, c, d, e;
  B(int a, int b, int c, int d, int e){
     this.a=a; this.b=b; this.c=c; this.d= d; this.e=e;
  }
}
class A {
    int a, b, c, d, e, f;
    A(int a, int b, int c, int d, int e, int f){
        this.a=a; this.b=b; this.c=c;
        this.d= d; this.e=e; this.f=f;
    }
}
class Demo {
    public static void main(String[] args) {
        A a = new A(3, 4, 5, 6, 7, 8);
        B b = new B(2, 1, 2, 3, 4);
        C c = new C(4, 5, 2);
    }
}
```

(b) What is the difference between method overloading and method overriding? Explain with appropriate example codes in Java. 6

(c) The same class may create different objects in different ways and under different conditions. OOP languages allow different constructors in a class to create different objects of the same class. Differentiate between the default constructor and explicit constructor of a class. What happens to the default constructor in Java if we write an explicit constructor in a class? 3

4. (a) (I) Why does Java not allow two classes to be inherited by the same class? Explain the problem (with appropriate example code) that may ensue if one class inherits two classes. 5+5

II) Explain in the light of your example given in your answer to (I) as to how using the interface avoids the problem.

(b) Explain the benefits of abstract classes to represent mixed-type components of an object. (For example, a document file like .docx or .odt file is an object with mixed-type components such as texts, images, lists, etc.) 4

5. (a) What are the two ways in Java a thread can be created? Why does Java offer these two ways instead of one only? 6

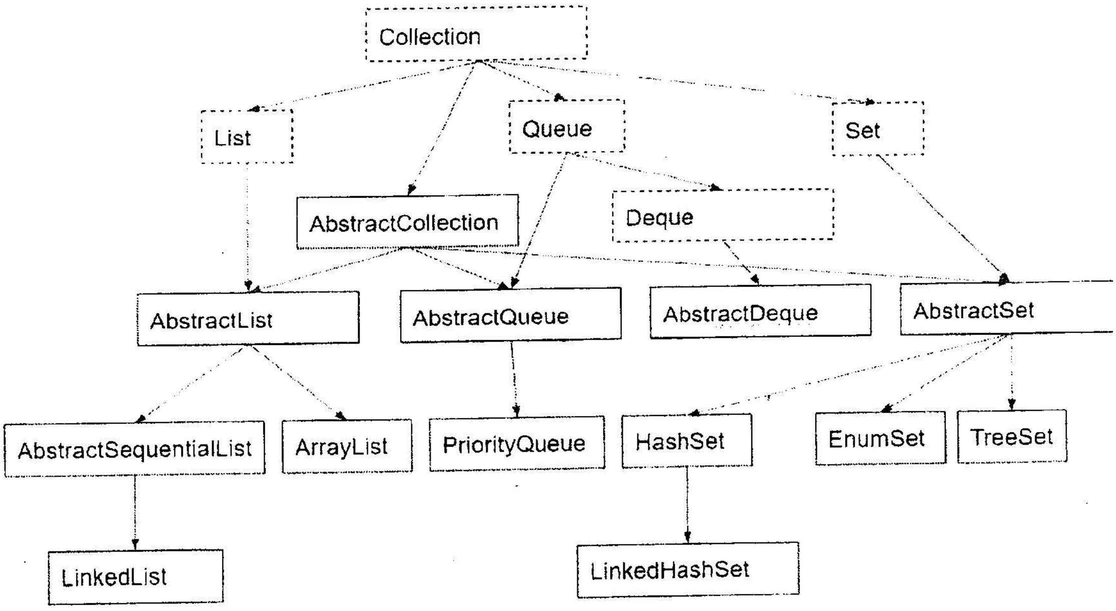
(b) Put appropriate exception handling code in the following code snippet. You must use the best practice of exception handling. 8

```
public static void main(String args[]{
 int i=0;
 FileInputStream fin= new FileInputStream(filename);
//may throw IOException

 do {
   i = fin.read();//may throw IOException
   if(i != -1) System.out.print((char) i);
 } while(i != -1);

 fin.close();//may throw IOException
}
```

6. (a) The following is the class hierarchy of Java's Collection framework. A dotted-lined box and solid-lined box indicate an interface and a concrete/abstract class, respectively. An arrow from one box to another indicates that the former box is inherited/implemented by the latter box. 3+3+4



I) You want to write a code that uses a HashSet data structure. What are the types of reference variables you can use?

II) Is the following code valid? Why or why not?

```
List L = new Collection();
L.add(56);
System.out.println(L);
```

III) Someone wrote the following code snippet:

```
ArrayList L1 = new ArrayList();
LinkedList L2 = new LinkedList();
if (user_input == 1) {
 L1.add(45);
 if (L1.get(0) == 45)
   System.out.println(“45 found.”);
}
else if (user_input == 2) {
 L2.add(45);
 if (L2.get(0) == 45)
   System.out.println(“45 found.”);
}
```

As you have probably guessed already, the code does not look like it was written by a professional OO programmer. Rewrite the code leveraging the class hierarchy so that the code size is minimized.

(b) Suppose your code may throw multiple exceptions E1, E2, E3 and E4 where E2 inherits E1, and E3 and E4 inherits E2. Write the try-catch ladder to catch each of these exceptions for your code. 4

7. (a) Suppose Language is a class that is inherited by three classes namely Bangla, English, and Arabic. Bangla class is further inherited by two classes: Cholti and Shadhu. For the sake of simplicity, assume that each of these classes has only a single constructor which takes a string as parameter and sets it to a local variable. 5

Write a generic class named AnyLang that works on only types of objects of Language class or any of its subclasses. AnyLang has only a single data structure which is an array of the object. In its constructor, it takes an array of objects as a parameter and assigns it to the local array reference variable. In another class called Demo, in the main function, create two objects of AnyLang class with Language and Bangla classes, respectively.

(b) Given the following code snippet, explain what is going on inside btn.setOnAction(...); expression: 5

```
public void start(Stage primaryStage) {
        primaryStage.setTitle("Hello World!");
        Button btn = new Button();
        btn.setText("Say 'Hello World'");
        btn.setOnAction(new EventHandler<ActionEvent>() {
            @Override
            public void handle(ActionEvent event) {
                System.out.println("Hello World!");
            }
        });
      //other code... ... ...
}
```

(c) Define the following two terms with an example of each: i) anonymous class, and (ii) functional interface. 2+2