

Bangladesh Army University of Science and Technology
Department of Computer Science and Engineering

Referred/Improvement/Backlog Examination, Fall 2018
Course No: CSE 2203
Time: 03 (Three) hours

Level-2 Term-II
Course Title: Object Oriented Programming II
Full Marks: 210

- N.B. (i) Answer any three questions from each PART (ii) Use separate answer script for each PART
 (iii) Marks allotted are indicated in the margin (iv) Symbols have their usual meanings

PART A

(Answer any **three** questions)

1. a) Consider the following java program and answer the questions.

(5x7)
=35

```
class A{
    A(){ System.out.println(" Default Constructor A "); }
    A(String n){ System.out.println(" Hello Java Parameterized const. "+n); }
    void show(){ System.out.println(" Hello A ");}
    void show(String n){ System.out.println(" Hello A "+n);}
}
class B extends A{
    B(String n){ System.out.println(" Hello Java Parameterized const. "+n); }
    void show(){ System.out.println(" Hello B "); }
    void show(int n){ System.out.println(" Hello B ");}
    void show(String n){System.out.println(" Hello B "+n);}
}
public class Access {
    public static void main(String args[]) {
        A a=new A();      a.show();      a.show("Shuvo");
        B b=new B("Sumi"); b.show();      b.show("Sumi");
    }
}
```

- i. What is the output of the program?
 - ii. How many classes and objects are there? What are they?
 - iii. How many default and parameterized constructors are there? What are they?
 - iv. Are there any method *overloading*? If so, write their name.
 - v. Are there any method *overriding*? If so, write their name.
 - vi. Is there any destructor in the above program? Why?
 - vii. Write the syntax to compile and run the above program.
2. a) What is bytecode? Why bytecode is called the magic code of Java? How JDK, JRE, JVM and bytecode are related with each other? Explain with appropriate figure and example. 10
- b) What will happen if we declare a variable, a method, and a class as final? Explain with a suitable code example. 10
- c) Consider the following sample java program and answer the following questions: 15
- i. In how many ways can the dimensions of the instance variable h , w , and d be set?
 - ii. Rewrite the program so that it will calculate the volume of a box object each time after the dimension is set.

<pre>class Box{ double h, w, d; void volume(){ System.out.println("Volume=" + (h*w*d)); } }</pre>	<pre>class Access{ public static void main(String args[]) { Box b=new Box(); a.volume(); // Each time after setting the dim. } }</pre>
--	---

3. a) What do you mean by *polymorphism* in java? Define compile time polymorphism and runtime polymorphism with suitable example. 10
- b) Define *dynamic method dispatch*? How can a superclass reference be used to access the members of subclass? Explain with example. 10

- c) Write down the flexibilities and restrictions of static variables and static method respectively. (10+)
What would be the output of the following code segment? [all the code reside within a class] =15

<pre>static int a = 2; static int b; static void meth(int x) { System.out.println("x = " + x); System.out.println("a = " + a); System.out.println("b = " + b); }</pre>	<pre>static{ a = a+3; b = a * 2; } static void init(int s){ a=s*10; b=b*100; meth(10); } public static void main(String args[]) { init (10); }</pre>
--	--

4. a) How is multiple inheritance achieved in java? Give an Example. What are the differences between abstract class and interface? (5+5)
=10
- b) Explain the following statement with example - "If you are extending an abstract class that has an abstract method, you must either provide the implementation of the method or make this class abstract." 10
- c) "Obtaining an object is a two steps process in java"- explain how? Draw the memory mapping (effect in memory) for the following statements. (5+10)
=15

<pre>class Customer{ long id; int age; double salary; }</pre>	<pre>Customer st1; st1= new Customer (); Customer st2= new Customer (); Customer st3=st2; st1= null; st3=st2=st1</pre>
---	--

PART B

(Answer any **three** questions)

5. a) What is *Daemon Thread*? Prove that the life of *Daemon Thread* depends on user threads. 10
- b) Suppose class A extends Thread class and class B implements Runnable interface. Now write the code to create a thread instance of both class and run the threads. 10
- c) Define the Life Cycle of a Thread. Briefly describe each state of thread life cycle with appropriate figure. Also mention the name of transition functions which are responsible to move the thread from state to state. 15

6. a) Consider the following Java code.

<pre>class Base{ public Base(){ public abstract void display(){ } }</pre>	<pre>class Child extends Base{ public Child(){ super(); public abstract void display(){ } }</pre>
---	---

(4+6)
=10

Answer the following:

- What is wrong in the above code? Explain briefly.
 - Propose changes in the above code for your expected output.
- b) Suppose that, class A and B resides in package p1, sp is a sub-package of p1 which hold class S. (15+10)
Under another package there is a class named C which is a subclass of A (C extends A) and include the main function. = 25
- Now answer the following questions-
- How many ways you can access the members of package p1 from package p2? Give example for each of them.
 - How can you access the resources of the subclass sp under p1 package? Write a short code to access the resource (say, a public function void *show()*) from p2 package.
7. a) Define checked and unchecked exception with example. Draw the figure of exception class hierarchy. (5+5)
=10

b). Identify the type of the following unchecked exceptions and define each of them.

(5x2)

=10

- i. `int a=50/0;`
- ii. `String s=null; System.out.println(s.length());`
- iii. `String s="abc"; int i=Integer.parseInt(s);`
- iv. `int a[]=new int[5]; a[10]=50;`
- v. `String[] s="abc"; int i= System.out.println(s[2]);`

c) Consider the following instances of String, StringBuilder and StringBuffer classes. Draw the memory mapping (stack, heap and constant pool) for each of them.

(1.5*10)

=15

1. `String st1 ="Sakib";`
2. `String st2 =" Al Hasan "; [space on both side]`
3. `String st3 ="Sakib";`
4. `String st4 =new String("Sakib");`
5. `String st5 =new String("Al Hasan ");`
6. `String st6 =new String(" Sakib ");`
7. `StringBuffer sb1 =new StringBuffer(" Sakib ");`
8. `StringBuffer sb2 =new StringBuffer(" Al Hasan ");`
9. `StringBuffer sb3 =new StringBuffer("Sakib ");`
10. `StringBuilder sb4 =new StringBuilder ("Al Hasan ");`

8. a) Define *functional interface* with example. What is the most common built in *functional interface* in java?

7

b) Consider the following sample program in java

(3x6)

=18

<pre>@FunctionalInterface // Annotation interface Cab{ void bookCab(); }</pre>	<pre>public class LambdaApp{ public static void main(String args[]){ // Todo code } }</pre>
--	---

Now answer the following:

- i. Develop a class that will implements the `bookCab()` method of the interface.
- ii. How to access the `bookCab()` method using polymorphic statement?
- iii. Give an anonymous class implementation of `bookCab()`.
- iv. How to use lambda expression to implements `bookCab()`?
- v. What will happen if we add another abstract or concrete method in the Cab interface?
- vi. Can we pass any argument in lambda function? If yes, explain how.

c) Define *generic* in java. How does generic reduce the code complexity in java? Give an example.

10