graphixs

BSCCS2005: Practice Assignment Questions with Solutions
Week {2}

{Write general instructions here}

1. What is the output of the following Java code? [MCQ:2points]

```
public class Example {
    public static void main(String[] args) {
        System.out.println("Addition of 10 and 20 is " + 10 + 20);
    }
}
```

◯ `Addition of 10 and 20 is 30`

√ `Addition of 10 and 20 is 1020`

◯ `"Addition of 10 and 20 is " + 10 + 20`

◯ `Error`

Solution: string concatenation is happening here.

- "Addition of 10 and 20 is " + 10 gives "Addition of 10 and 20 is 10" here string is concatenated with an integer which results in a string.
- "Addition of 10 and 20 is 10"+ 20 gives "Addition of 10 and 20 is 1020" here string is concatenated with an integer which results in a string.

2. What is the output of the following Java code? [MCQ:2points]

```
public class FClass{
    public static void main(String[] args) {
        int[] a = {1, 2, 3};
        int[] b = a;
        System.out.print(a == b);
    }
}
```

◯ `a == b`

◯ `false`

√ `true`

◯ `{1, 2, 3} == {1, 2, 3}`

Solution: It prints true. The `==` operator compares whether `a` and `b` are referring to the same memory location.

3. What is the output of the following Java code? [MCQ:2points]

```
public class FClass{
    public static void main(String[] args) {
        String[] str1, str2 = {"apples", "grapes", "bananas"};
        System.out.print(str1[1]);
    }
}
```

◯ "apples"

◯ "grapes"

◯ {"apples"}

◯ {"grapes"}

√ **Compiler Error**

Solution: `str1` is declared but not initialized. It gives an error because we are trying to access `str1[1]` which is not initialized

4. Consider the Java program below. [MCQ:2 points]

```
class FClass{
    public static void main(String[] args) {
        int i1 = 10, i2 = 29;
        double d;
        d = i1 / i2 * i2;
        System.out.print(d + " ");
        d = i2 * i1 / i2;
        System.out.print(d + " ");
    }
}
```

What will be the output?

- ◯ `0.0 0.0`
- ◯ `10.0 10.0`
- ◯ `10.0 0.0`
- √ `0.0 10.0`

Solution: The computations for statement `d = i1 / i2 * i2;` take place as follows:
Since `i1` and `i2` both are of type `int`, `i1 / i2` is an integer division which results in 0. So `0 * i2 = 0`. But `d` is of type `double` so result would be displayed as `0.0`.
The part of the statement `i2 * i1` would be computed as `290` which is an `int`. Next, `290 / i2` is an integer division which would result as `10`, Finally, the result is assigned to `d`. Since `d` is of type `double`, it would be printed as `10.0`.

5. Consider the Java program below. [MCQ:2 points]

```
class FClass{
    int i;
    public static void main(String[] args) {
        final int i = 10;
        i = 20;
        System.out.println(i);
    }
}
```

Choose the correct option regarding the given program.

- √ **The program generates a compilation error.**
- ◯ The program generates a run time error.
- ◯ The program generates output: 10
- ◯ The program generates output: 20

Solution: `i` is declared as `final`. Thus, it cannot be reassigned.

6. Consider the Java program below. [MCQ:2 points]

```
class SClass{
    int i = 10;
    public SClass(int _i) { i = 20; }
    public void print() { System.out.println(i); }
}
class PQ{
    public static void main(String[] args) {
        int i = 30;
        SClass obj = new SClass(i);
        obj.print();
    }
}
```

Choose the correct option regarding the given program.

- ◯ The program generates a compilation error.
- ◯ The program generates output: 10
- √ **The program generates output: 20**
- ◯ The program generates output: 30

Solution: The constructor:
`public SClass(int _i) { i = 20; }`
initializes `i` to 20. Thus, it prints 20.

7. Consider the Java program below. [MCQ:2 points]

```
class StringExample {
    public static void main(String[] args) {
        String str = "we are here to learn java";
        String str1 = "";
        char[] ch = {‘a’, ‘e’, ‘i’,‘o’, ‘u’};
        for(char c : ch){
            for (int i = 0; i < str.length(); i++){
                if (str.charAt(i) == c){
                    str1 += c;
                }
            }
        }
    }
}
```

What will be the value of `str1` at the end of execution?

- ◯ `"eaeeeoeaaa"`
- ◯ `"wrhrtlrnjv"`
- √ `"aaaaeeeeeo"`
- ◯ `"aeiou"`

Solution:

- Program execution starts from main function
- `str1` is assigned to a string, `str2` is an empty string.
- `ch` is a character array, and is initialized to {’a’, ‘e’, ‘i’, ‘o’, ‘u’}
- Each element of the character array is compared with each and every element of `str`. if both these elements are same, then that element is concatenated, to `str1`.
- Hence `str1` will have all the vowels present in `str` and are arranged in alphabetical order.

8. Consider the Java code given below. [MCQ : 2 points]

```
import java.util.Scanner;
class SwitchEx
{
    public static void main(String[] args)
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Please enter the any ASIA cricket team");
        String country=sc.next();
        switch(country)
        {
            case "India":
                System.out.println("Delhi is capital of "+country);
            case "SriLanka":
                System.out.println("Colombo is capital of "+country);
            default:
                System.out.println("Please enter valid country name:");
                break;
        }
    }
}
```

If we provide `SriLanka` as the input to the program, what will be the output?

- ◯ Compilation fails.
- ◯ `Colombo is capital of India.`
- ◯ `Delhi is capital of India.`
  `Colombo is capital of SriLanka.`
  `Please enter valid country name:`
- √ `Colombo is capital of SriLanka`
  `Please enter valid country name:`

Solution: If you give `SriLanka` as input to the above program, it matches to the second case and executes corresponding statements, there is no break used after the case, it will execute default also.

9. Consider the Java code given below. [MCQ : 2 points]

```
class  ForComma
{
    public static void main(String[] args)
    {
        int j=20;
        for (int i=10;i<j ;i++,j--)    //LINE-1
        {
            System.out.println("Hello World!");
        }
    }
}
```

Choose the correct option regarding the given code.

◯ Compilation fails due to error at `LINE-1`.

◯ `Hello World!` printed 9 times.

√ `Hello World!` **printed 5 times.**

◯ `Hello World!` printed until stack overflow.

Solution:

- First iteration `j=20, i=10, i<j => true`,
  - Thus `Hello World!` will be printed.
- Second iteration `j=19, i=11, i<j => true`
  - Thus `Hello World!` will be printed.
- Third iteration `j=18, i=12, i<j => true`,
  - Thus `Hello World!` will be printed.
- Fourth iteration `j=17, i=13, i<j => true`,
  - Thus `Hello World!` will be printed.
- Fifth iteration `j=16, i=14, i<j => true`

- Thus `Hello World!` will be printed.

- Sixth iteration `j=15, i=10, i<j => false`,
  - Thus loop will be terminated.

10. Consider the Java code given below. [MCQ : 2 points]

```
class  ForEx
{
    public static void main(String[] args)
    {
        for (;;)
        {
            System.out.println("Hello World!");
        }
    }
}
```

Choose the correct option regarding the given code.

- ◯ The code generates a compilation error.
- ◯ The code generates a run time error.
- √ `Hello World!` **would be printed forever.**
- ◯ Nothing will be printed.

Solution: No error in the code, `Hello world!` message would be printed infinite times as there is no exit condition for the given *for* loop.

11. Consider the Java code given below. [MCQ : 2 points]

```
public class Demo
{
    public static void main(String args[])
    {
        int x=10;
        int y=20;
        y=+x;
        System.out.println(y);
    }
}
```

what will be the output?

- ◯ compilation failed.
- √ 10
- ◯ 20
- ◯ 30

Solution: `10` will be printed as output.
`y=+x;` this statement will assign the `x` value to the `y`, `y=+x` and `y=x` both are same.

12. Consider the Java code below. [MCQ:2 points]

```
class FClass
{
    public static void main(String apple[])
    {
        int a = 34.0;
        int b = 7;
        int c = a % b;
        System.out.println(c);
    }
}
```

Answer the following.

(a) The above code does not compile because the name of the String array is not proper.

◯ True

√ **False**

(b) The above code gives 6 as the output.

◯ True

√ **False**

(c) The above code does not compile because the data types are mismatched.

√ **True**

◯ False

Solution:

Inside the `main()` method a variable `a` is declared which is of type `int`. So the value that `a` can hold any integer value. But `a` is assigned to `34.0` which is of type `double`. This leads to mismatch of type. So compilation error occurs.

To make this code work we need to explicitly typecast it because the double data type is larger than int. The proper version of this code yields `6` as an output.

```
class FClass
{
    public static void main(String apple[])
    {
        int a = (int)34.0;
        int b = 7;
        int c = a % b;
        System.out.println(c);
    }
}
```

13. Answer the following. [MCQ:2 points]

(a) The following is a legal array declaration.

```
String[] State = new String[] {"Chennai", "Kolkata", "Tripura" };
```

√ **True**

◯ False

(b) Arrays are objects

√ **True**

◯ False

(c) An array index runs from `1` to `a.length`

◯ True

√ **False**

(d) Size of an array expands automatically when it is full.

◯ True

√ **False**

Solution:
An array index runs from `0` to `a.length-1`.
Size of an array can not expand automatically.

14. Considering the following Java code, choose all the correct statement/(s) from among the given options regarding this code.

[MSQ:2points]

```
class Employee
{
    int eid;
    String ename;

    public Employee(Employee e)
    {
        this.eid = e.eid;
        this.ename = e.ename;
    }
    public void display()
    {
        System.out.println("eid: "+eid);
        System.out.println("ename: "+ename);
    }
}

class Fclass
{
    public static void main(String[] args)
    {
        Employee e1 = new Employee();
        e1.display();
    }
}
```

- √ **One can't create an object of the *Employee* class.**
- ◯ Program runs successfully, and the data members `eid` and `ename` contains random garbage value.
- ◯ Program runs successfully, and the data members `eid` and `ename` contains *0* and *null* respectively.
- √ **This code results in a compilation error.**

Solution:
In the above code a copy constructor is defined and therefore the default no argument constructor is hidden. Now the code has neither the default constructor nor a explicitly defined constructor for creating a parameterized or non-parameterized

object. So **we can't create an object of the Employee class.**
Therefore the statement `Employee e1 = new Employee();`, creates a compile time error.

15. What will be the output of the following Java code?

[MCQ:2points]

```
import java.util.*;

class Employee
{
    int eid;
    String ename;
    String eprojects[];

    public Employee(Employee e)
    {
        this.eid = e.eid;
        this.ename = e.ename;
        this.eprojects = e.eprojects;
    }
    public Employee(int id, String name, String[] projects)
    {
        this.eid = id;
        this.ename = name;
        this.eprojects = projects;
    }
    public void display()
    {
        System.out.println("eid: "+eid);
        System.out.println("ename: "+ename);
        System.out.println("eprojetcs: ");
        for(String s : eprojects){
            System.out.println(s);
        }
    }
    public void mutator()
    {
        this.ename = this.ename + "__IITM";
        this.eprojects[0] = "P6";
    }

}
public class FClass
{
    public static void main(String[] args)
    {
```

```
        String arr[] = {"P1","P2"};
        Employee e1 = new Employee(12,"Alex",arr);
        Employee e2 = new Employee(e1);
        e1.mutator();
        e1.display();
        e2.display();
    }
}
```

◯ This program results in a compilation error.

◯ eid: 12
ename: Alex__IITM
eprojetcs:
P6
P2
eid: 12
ename: Alex__IITM
eprojetcs:
P6
P2

◯ eid: 12
ename: Alex__IITM
eprojetcs:
P6
P2
eid: 12
ename: Alex
eprojetcs:
P1
P2

√ **eid: 12**
ename: Alex__IITM
eprojetcs:
P6
P2
eid: 12
ename: Alex
eprojetcs:
P6
P2

Solution:
The statement `this.eprojects = e.eprojects;` in the copy constructor does a shallow copy of the eprojects array therefore when the array is mutated for object *e1* the same change is reflected for instance *e2* as well.
Will be discussed in the live session.