

Exam 2: QUESTIONS

Question 1 of 8

What will be the output if you run the following program?

```
public class TestClass
{
    public static void main(String args[])
    {
        int i;
        int j;
        for (i = 0, j = 0 ; j < 1 ; ++j , i++)
        {
            System.out.println( i + " " + j );
        }
        System.out.println( i + " " + j );
    }
}
```

Select 1 correct option.

- a** ☐ 0 0 will be printed twice.
- b** ☐ 1 1 will be printed once.
- c** ☐ 0 1 will be printed followed by 1 2.
- d** ☐ 0 0 will be printed followed by 1 1.
- e** ☐ It will print 0 0 and then 0 1.

Question 2 of 8

What will be the result of attempting to compile and run the following class?

```
public class TestClass
{
    public static void main(String args[ ] )
    {
        int i = 1;
        int[] iArr = {1};
        incr(i) ;
        incr(iArr) ;
        System.out.println( "i = " + i + "  iArr[0] = " + iArr [ 0 ] ) ;
    }
    public static void incr(int    n ) { n++ ; }
    public static void incr(int[ ] n ) { n [ 0 ]++ ; }
}
```

Select 1 correct option.

- a ☐ The code will print i = 1 iArr[0] = 1;
- b ☐ The code will print i = 1 iArr[0] = 2;
- c ☐ The code will print i = 2 iArr[0] = 1;
- d ☐ The code will print i = 2 iArr[0] = 2;
- e ☐ The code will not compile.

Question 3 of 8

Which of the following code snippets will print exactly 10?

1.

```
Object t = new Integer(106);  
int k = ((Integer) t).intValue()/10;  
System.out.println(k);
```
2.

```
System.out.println(100/9.9);
```
3.

```
System.out.println(100/10.0);
```
4.

```
System.out.println(100/10);
```
5.

```
System.out.println(3 + 100/10*2-13);
```

Select 3 correct options

- a ☐ 1
- b ☐ 2
- c ☐ 3
- d ☐ 4
- e ☐ 5

Question 4 of 8

What do you mean by "encapsulation" ?

Select 1 correct option.

- a** ☐ There is no way to access member variable.
- b** ☐ There are no member variables.
- c** ☐ Member fields are declared private but public accessor/mutator methods are provided to access and change their values.
- d** ☐ Data fields are declared public and accessor methods are provided to access and change their values.
- e** ☐ None of the above.

Question 5 of 8

Consider the following array definitions:

```
int[] array1, array2[];  
int[][] array3;  
int[] array4[], array5[];
```

Which of the following are valid statements?

Select 3 correct options

- a** ☐ array2 = array3;
- b** ☐ array2 = array4;
- c** ☐ array1 = array2;
- d** ☐ array4 = array1;
- e** ☐ array5 = array3

Question 6 of 8

Which of the following code fragments will successfully initialize a two-dimensional array of chars named cA with a size such that cA[2][3] refers to a valid element?

1.
`char[][] cA = { { 'a', 'b', 'c' }, { 'a', 'b', 'c' } };`
2.
`char cA[][] = new char[3][];
for (int i=0; i<cA.length; i++) cA[i] = new char[4];`
3.
`char cA[][] = { new char[] { 'a', 'b', 'c' } , new char[] { 'a', 'b', 'c' } };`
- 4
`char cA[3][2] = new char[][] { { 'a', 'b', 'c' }, { 'a', 'b', 'c' } };`
5.
`char[][] cA = { "1234", "1234", "1234" };`

Select 1 correct option.

- a ☐ 1, 3
- b ☐ 4, 5
- c ☐ 2, 3
- d ☐ 1, 2, 3
- e ☐ 2

Question 7 of 8

What will be printed by the following code if it is run with command line: java TestClass -0.50 ?

```
public class TestClass
{
    public static double getSwitch(String str)
    {
        return Double.parseDouble(str.substring(1, str.length()-1) );
    }
    public static void main(String args [])
    {
        switch(getSwitch(args[0]))
        {
            case 0.0 : System.out.println("Hello");
            case 1.0 : System.out.println("World"); break;
            default : System.out.println("Good Bye");
        }
    }
}
```

Select 1 correct option.

- a ☐ Hello
- b ☐ World
- c ☐ Hello World
- d ☐ Hello World Good Bye
- e ☐ None of the above.

Question 8 of 20

What will the following program print?

```
class Test
{
    public static void main(String args[])
    {
        int c = 0;
        boolean flag = true;
        for(int i = 0; i < 3; i++)
        {
            while(flag)
            {
                c++;
                if(i>c || c>5) flag = false;
            }
        }
        System.out.println(c);
    }
}
```

Select 1 correct option.

- a ☐ 3
- b ☐ 4
- c ☐ 5
- d ☐ 6
- e ☐ 7

SOLUTIONS

Question 1 of 8

What will be the output if you run the following program?

```
public class TestClass
{
    public static void main(String args[])
    {
        int i;
        int j;
        for (i = 0, j = 0 ; j < 1 ; ++j , i++)
        {
            System.out.println( i + " " + j );
        }
        System.out.println( i + " " + j );
    }
}
```

Select 1 correct option.

- a ☐ 0 0 will be printed twice.
- b ☐ 1 1 will be printed once.
- c ☐ 0 1 will be printed followed by 1 2.
- d ☒ 0 0 will be printed followed by 1 1.**
- e ☐ It will print 0 0 and then 0 1.

General Comments
j will be less than 1 for only first iteration. So, first it'll print 0, 0. Next, i and j are incremented. prefix and postfix operators don't matter in this case as the increment portion of for is preformed after the iteration. After the first increment ++j and i++ will produce the same effect. And after the end of the loop i and j will have same values (ie. 1). Now, the loop will not execute as j is not less than 1 at the start of the loop so the condition fails and it comes out of the loop. Finally, it will print 1,1.

Question 2 of 8

What will be the result of attempting to compile and run the following class?

```
public class TestClass
{
    public static void main(String args[ ] )
    {
        int i = 1;
        int[] iArr = {1};
        incr(i) ;
        incr(iArr) ;
        System.out.println( "i = " + i + " iArr[0] = " + iArr [ 0 ] ) ;
    }
    public static void incr(int    n ) { n++ ; }
    public static void incr(int[ ] n ) { n [ 0 ]++ ; }
}
```

Select 1 correct option.

- a ☐ The code will print i = 1 iArr[0] = 1;
- b ☒ The code will print i = 1 iArr[0] = 2;**
- c ☐ The code will print i = 2 iArr[0] = 1;
- d ☐ The code will print i = 2 iArr[0] = 2;
- e ☐ The code will not compile.

General Comments

Arrays are proper objects (ie. iArr instanceof Object is true) and Object's references are passed by value.

So the value of reference of iArr is passed to the method incr(int[] i); This method changes the actual value of the int element at 0.

Question 3 of 8

Which of the following code snippets will print exactly 10?

1.

```
Object t = new Integer(106);  
int k = ((Integer) t).intValue()/10;  
System.out.println(k);
```
2.

```
System.out.println(100/9.9);
```
3.

```
System.out.println(100/10.0);
```
4.

```
System.out.println(100/10);
```
5.

```
System.out.println(3 + 100/10*2-13);
```

Select 3 correct options

- a** ☒ 1
- b** ☐ 2
- c** ☐ 3
- d** ☒ 4
- e** ☒ 5

General Comments

1. `int k = ((Integer) t).intValue()/10;`

Since both the operands of `/` are ints, it is a integer division. This means the resulting value is truncated (and not rounded). Therefore, the above statement will print 10 and not 11.

`3 + 100/10*2-13` will be parsed as: `3 + (100/10)*2-13`. This is because the precedence of `/` and `*` is same and since the expression is evaluated from left to right, the operands are grouped on first come first served basis. [This is not the right terminology but you will be able to answer the questions if you remember this rule.]

Question 4 of 8

What do you mean by "encapsulation" ?

Select 1 correct option.

- a** ☐ There is no way to access member variable.
- b** ☐ There are no member variables.
- c** ☒ Member fields are declared private but public accessor/mutator methods are provided to access and change their values.
- d** ☐ Data fields are declared public and accessor methods are provided to access and change their values.
- e** ☐ None of the above.

Question 5 of 8

Consider the following array definitions:

```
int[] array1, array2[];  
int[][] array3;  
int[] array4[], array5[];
```

Which of the following are valid statements?

Select 3 correct options

a ☒ array2 = array3;

b ☒ array2 = array4;

c ☐ array1 = array2;

d ☐ array4 = array1;

e ☒ array5 = array3

General Comments

There is a subtle difference between: `int[] i;` and `int i[];` although in both the cases, `i` is an array of integers.

The difference is, if you declare multiple variables in the same statement such as: `int[] i, j;` and `int i[], j;`, `j` are not of the same type.

In the first case, `j` is an array of integers while in the second case, `j` is just an integer.

Therefore, in this question:

`array1` = array of `int`

`array2`, `array3`, `array4`, and `array5` = array of array `int`

Therefore, option 1, 2 and 5 are valid.

Question 6 of 8

Which of the following code fragments will successfully initialize a two-dimensional array of chars named cA with a size such that cA[2][3] refers to a valid element?

1.
`char[][] cA = { { 'a', 'b', 'c' }, { 'a', 'b', 'c' } };`
2.
`char cA[][] = new char[3][];
for (int i=0; i<cA.length; i++) cA[i] = new char[4];`
3.
`char cA[][] = { new char[]{ 'a', 'b', 'c' } , new char[]{ 'a', 'b', 'c' } };`
- 4
`char cA[3][2] = new char[][] { { 'a', 'b', 'c' }, { 'a', 'b', 'c' } };`
5.
`char[][] cA = { "1234", "1234", "1234" };`

Select 1 correct option.

- a** ☐ 1, 3
- b** ☐ 4, 5
- c** ☐ 2, 3
- d** ☐ 1, 2, 3
- e** ☒ 2

General Comments

1 and 3 declare a two dimensional array alright but they create the array of size 2, 3. And cA[2][3] means we need an array of size 3, 4 because the numbering starts from 0.

4 : You cannot put array size information on LHS.

5. This is a one dimensional array and that too of strings. Note that a java String is not equivalent to 1 dimensional char array. This leaves us with only one choice 2

Question 7 of 8

What will be printed by the following code if it is run with command line: java TestClass -0.50 ?

```
public class TestClass
{
    public static double getSwitch(String str)
    {
        return Double.parseDouble(str.substring(1, str.length()-1) );
    }
    public static void main(String args [])
    {
        switch(getSwitch(args[0]))
        {
            case 0.0 : System.out.println("Hello");
            case 1.0 : System.out.println("World"); break;
            default : System.out.println("Good Bye");
        }
    }
}
```

Select 1 correct option.

- a ☐ Hello
- b ☐ World
- c ☐ Hello World
- d ☐ Hello World Good Bye
- e ☒ None of the above.

General Comments

The program will not even compile because double/float/long cannot be used in switch(...).

Question 8 of 8

What will the following program print?

```
class Test
{
    public static void main(String args[])
    {
        int c = 0;
        boolean flag = true;
        for(int i = 0; i < 3; i++)
        {
            while(flag)
            {
                c++;
                if(i>c || c>5) flag = false;
            }
        }
        System.out.println(c);
    }
}
```

Select 1 correct option.

- a ☐ 3
- b ☐ 4
- c ☐ 5
- d ☒ 6**
- e ☐ 7

General Comments



In the first iteration of for loop, the while loop keeps running till c becomes 6. Now, for all next for loop iteration, the while loop never runs as the flag is false. So final value of c is 6