B.Sc. (Hons.) Computer Science

JAVA PROGRAMMING Question Bank

Unique Paper Code: 32341201

1. What will be the output of the program?

```
public class Foo
{
    public static void main(String[] args)
    {
        try
        {
            return;
        }
        finally
        {
            System.out.println( "Finally" );
        }
    }
}
```

A. Finally

B. Compilation fails.

C. The code runs with no output.

D. An exception is thrown at runtime.

2. What will be the output of the program?

```
try
{
    int x = 0;
    int y = 5 / x;
}
catch (Exception e)
{
    System.out.println("Exception");
}
catch (ArithmeticException ae)
{
    System.out.println(" Arithmetic Exception");
}
System.out.println("finished");
```

A. Finished

B. Exception

C. Compilation fails.

D. Arithmetic Exception

3.

```
interface DoMath
{
    double getArea(int rad);
}
```

```
interface MathPlus
{
   double getVol(int b, int h);
}
/* Missing Statements ? */
```

which two code fragments inserted at end of the program, will allow to compile?

1. class AllMath extends DoMath { double getArea(int r); }
2. interface AllMath implements MathPlus { double getVol(int x, int y); }
3. interface AllMath extends DoMath { float getAvg(int h, int l); }
4. class AllMath implements MathPlus { double getArea(int rad); }
5. abstract class AllMath implements DoMath, MathPlus { public double getArea(int rad) { return rad * rad * 3.14; } }

A. 1 only

B. 2 only

C. 3 and 5

D. 1 and 4

4. What will be the output of the program?

```
public class X
{
   public static void main(String [] args)
   {
      try
      {
         badMethod();
         System.out.print("A");
      }
      catch (RuntimeException ex) /* Line 10 */
      {
         System.out.print("B");
      }
      catch (Exception ex1)
      {
         System.out.print("C");
      }
      finally
      {
         System.out.print("D");
      }
      System.out.print("E");
   }
   public static void badMethod()
```

```
    {
        throw new RuntimeException();
    }
}
```

A. BD **B.** BCD

C. BDE **D.** BCDE

5. Which three statements are true?

1. The default constructor initializes method variables.
2. The default constructor has the same access as its class.
3. The default constructor invokes the no-arg constructor of the superclass.
4. If a class lacks a no-arg constructor, the compiler always creates a default constructor.
5. The compiler creates a default constructor only when there are no other constructors for the class.

A. 1, 2 and 4 **B.** 2, 3 and 5

C. 3, 4 and 5 **D.** 1, 2 and 3

6. What will be the output of the program?

```
public class Test
{
    public static void aMethod() throws Exception
    {
        try /* Line 5 */
        {
            throw new Exception(); /* Line 7 */
        }
        finally /* Line 9 */
        {
            System.out.print("finally "); /* Line 11 */
        }
    }
    public static void main(String args[])
    {
        try
        {
            aMethod();
        }
        catch (Exception e) /* Line 20 */
        {
            System.out.print("exception ");
        }
        System.out.print("finished"); /* Line 24 */
    }
}
```

A. finally

B. exception finished

C. finally exception finished

D. Compilation fails

7. What will be the output of the program?

```
public class X
{
    public static void main(String [] args)
    {
        try
        {
            badMethod();
            System.out.print("A");
        }
        catch (Exception ex)
        {
            System.out.print("B");
        }
        finally
        {
            System.out.print("C");
        }
        System.out.print("D");
    }
    public static void badMethod() {}
}
```

A. AC

B. BC

C. ACD

D. ABCD

8. What will be the output of the program?

```
public class X
{
    public static void main(String [] args)
    {
        try
        {
            badMethod(); /* Line 7 */
            System.out.print("A");
        }
        catch (Exception ex) /* Line 10 */
        {
            System.out.print("B"); /* Line 12 */
        }
        finally /* Line 14 */
        {
            System.out.print("C"); /* Line 16 */
        }
        System.out.print("D"); /* Line 18 */
    }
    public static void badMethod()
    {
        throw new RuntimeException();
    }
}
```

A. AB **B.** BC

C. ABC **D.** BCD

9. What will be the output of the program?

```
public class A
{
    void A() /* Line 3 */
    {
        System.out.println("Class A");
    }
    public static void main(String[] args)
    {
        new A();
    }
}
```

1. Class A
2. Compilation fails.
3. An exception is thrown at line 3.
4. The code executes with no output.

10. What will be the output of the program ?

```
public class Test
{
    public static void main(String [] args)
    {
        signed int x = 10;
        for (int y=0; y<5; y++, x--)
            System.out.print(x + ", ");
    }
}
```

A. 10, 9, 8, 7, 6,

B. 9, 8, 7, 6, 5,

C. Compilation fails.

D. An exception is thrown at runtime

```
11. public void test(int x)
{
    int odd = 1;
    if(odd) /* Line 4 */
    {
        System.out.println("odd");
    }
    else
    {
        System.out.println("even");
    }
}
```

Which statement is true?

A. Compilation fails.

B. "odd" will always be output.

C. "even" will always be output.

D. "odd" will be output for odd values of x, and "even" for even values.

12.

What will be the output of the program?

```
interface Count
{
    short counter = 0;
    void countUp();
}
public class TestCount implements Count
{
    public static void main(String [] args)
    {
        TestCount t = new TestCount();
        t.countUp();
    }
    public void countUp()
    {
        for (int x = 6; x>counter; x--, ++counter) /* Line 14 */
        {
            System.out.print(" " + counter);
        }

    }}
```

A. 0 1 2 B. 1 2 3

C. 0 1 2 3 D. 1 2 3 4

E. Compilation fails