1

Given the following definition:

```
int len = 4;
    int[][] matrix = new int[len][len];
```

Consider the following code:

```
int total = 0;
for (int row = 0; row < len; row++)
{
        total = total + matrix[row][1];
}
```

Assume that `matrix` has the following values. Note that `matrix[0][2]` is 6.

4	5	6	7
0	1	2	3
3	2	1	0
8	9	1	2

What will the value of `total` be after the code executes?

- 10
- 15
- 17
- 12
- 22

2

What is the difference between an interface and an abstract class?

**QUESTION WRITTEN ABOUT PRE JAVA VERSION 8!

- There is no difference.
- Abstract classes can have methods with bodies (code in them), but interfaces can not.
- Abstract classes can be instantiated, while interfaces can not.
- Abstract classes can be extended, but interfaces can not.
- Abstract classes can declare abstract methods, but interfaces can not.

3

Consider the following field and method declarations.

```
private List list<Integer>;
        public void mystery(int n)
    {
        for (int i= 0; i < n; i++)
        {
            Object obj = list.remove(0);
            list.add((Integer)obj);
       }
    }
```

Assume that `list` has been initialized with the following `Integer` objects:
[9, 3, 17, 2, 16, 4, 1]

Which of the following shows the values in `list` after a call to `mystery(4)`?

- [9, 3, 17, 2, 16, 4, 1]
- [1, 4, 16, 2, 17, 3, 9]
- [9, 3, 17, 16, 4, 1, 2]
- [16, 4, 1, 9, 3, 17, 2]
- [2, 16, 4, 1, 9, 3, 17]

4

Consider the following method:

```
public static int mystery(int y)
    {
      y = 2 * y + y;
      y = 2 * y + y;
      return y;
    }
```

Which of the following expressions can be use to replace the body of mystery so that mystery will return the same result for all values of `y`?

- `return 9 * y;`
- `return 6 * y;`
- `return y;`
- `return 3 * y;`
- `return 4 * y;`

5

Consider the following method:

```
public static int mystery(int y)
    {
      y = 2 * y + y;
      y = 2 * y + y;
      return y;
    }
```

Which of the following expressions can be use to replace the body of mystery so that mystery will return the same result for all values of `y`?

- `return 9 * y;`
- `return 6 * y;`
- `return y;`
- `return 3 * y;`
- `return 4 * y;`

6

Class C extends class B, which extends class A. Also, all of the three classes implement a public method test(). How can a method in an object of class C invoke the test() method defined in class A (without creating a new instance of class A)?

- test();
- super.super.test();
- super.test();
- this.test();
- There is no way to call a method in a grandparent class from a grandchild class.

7

Given the following declarations.

```
public class Vechicle {
      public void test(Car x, SportsCar y) {}
    }

   public class Car extends Vechicle {
    }

    public class SportsCar extends Car {
    }
```

Also consider the following code that appears in a different class.

```
Vechicle v = new Vechicle();
    Car c = new Car();
    SportsCar sporty = new SportsCar();
```

Which of the following is a correct call to `test`?

- `v.test(sporty,v);`
- `sporty.test(c,c);`
- `v.test(sporty,c);`
- `sporty.test(sporty,v);`
- `c.test(sporty,sporty);`

8

Consider the following method.

```
public static int recur(int n) {
        if (n <= 1) return 1;
        else return (recur(n-1) + recur(n-2));
    }
```

What value is returned from `recur(5)`?

- 8
- 1
- 2
- 5
- 3

9

Consider the following code segment.

```
for (int k = 0; k < 20; k = k + 1)
{
  if (k % 2 == 0)
    System.out.print(k + " ");
}
```

What is printed as a result of executing the code segment?

- 1 3 5 7 9 11 13 15 17 19
- 0 2 4 6 8 10 12 14 16 18
- 2 4 6 8 10 12 14 16 18
- 3 6 9 12 15 18
- 0 2 4 6 8 10 13 14 16 18 20

10

Consider the following code segment.

```
List<String> list = new ArrayList<String>();
list.add("a");
list.add("b");
list.set(1,"c");
list.add(2, "d");
list.set(2, "e");
list.add("g");
System.out.println(list);
```

What is printed as a result of executing the code segment?

- [a, c, e, d, g]
- [c, e, d, b, g]
- [a, c, e, g]
- [a, b, e, d, g]
- [a, c, e, d, b, g]

11

Given the following class declarations. Assume that `Parent p = new Child();` appears in a client program. What is the result of the call `p.m1()`?

```
public class Parent {
    public void m1() {
          System.out.print("pm1");
          m2();
    }
    public void m2() {
          System.out.print("pm2");
    }
}

public class Child extends Parent {
      public void m1()
      {
         super.m1();
         System.out.print("cm1");
      }
      public void m2()
      {
         super.m2();
         System.out.print("cm2");
      }
}
```

- pm1pm2cm2cm1
- pm1pm2
- pm1pm2cm1cm2
- pm1cm1
- pm1

12

Given the following class declarations.

```
public class Animal {
       // constructors not shown
       public void eat()
{ // code not shown
}
   }

   public class Bear extends Animal {
      // constructors not shown
      public void growl()
     { // code not shown
     }
   }
```

Assume that the following declaration is in a different class.

```
Animal b = new Bear();
```

Which of the following will compile without error?

```
I.      b.eat();
II.      b.growl;
III.       ((Bear) b).growl();
```

- I only
- II only
- III only
- I and III only
- I, II, and III

13

Question 13 (1 point)

Given the following method and what would the result be when m is executed?

```
public void m(int[][]p)
{
  int height = p.length;
  for (int row = 0; row < height / 2; row++)
  {
    for (int col = 0; col <p[0].length; col++)
   {
     p[row][col] = p[height - row - 1][col];
    }
  }
}
```

- Copies the values from the top half to the bottom half of the 2D array
- Copies the values from the left halt to the right half of the 2D array
- Copies the values from the bottom half to the top half of the 2D array
- Copies the values from the right half to the left half of the 2D array
- All values remain the same.

14

What is the output from mystery(4321) when mystery is defined as follows:

```
//precondition: x >=0
public static void mystery (int x) {
  System.out.print(x % 10);

  if ((x / 10) != 0) {
    mystery(x / 10);
  }
  System.out.print(x % 10);
}
```

- 12344321
- 1234
- 4321
- 43211234
- 32144123

15

A classroom is a room and a building has many rooms. If the three classes `Room`, `Classroom`, and `Building` create objects that have the same relationship which of the following is the most appropriate set of declarations?

- `public class Room extends Classroom implements i.  Building { … }`
- `public class Classroom extends Room { … }`
  `public class Building`
  `{ private Room[] rooms; …. }`
- `public class Room extends Building`
  `{ private Classroom room; …. }`
- `public class Classroom extends Building, Room { … }`
- `public class Room extends Classroom, Building { … }`

16

Given the following code which of the answers best describes the conditions needed for `temp` to be true when it is returned?

```
boolean temp = false;
int count = 0;
for ( int testVal : a)
{
  if ( testVal == val ) count++;
}
temp = count > 1;
return temp;
```

- Whenever the first element in `a` is equal to `val`
- Whenever `a` contains any element which equals `val`.
- Whenever more than 1 element in `a` is equal to `val`.
- Whenever exactly 1 element in `a` is equal to `val`.
- Whenever the last element in `a` is equal to `val`.

17

What is the output from the following code segment?

```
for (int j = 1; j <=5; j++) {
  for (int k = 1; k < 3; k++)
      System.out.print(j * k + " ");
}
```

- 1 1 1 2 2 1 2 2 3 1 3 2 4 1 4 2 5 1 5 2
- 1 2 2 4 3 6 4 8
- 1 1 1 2 2 1 2 2 3 1 3 2 4 1 4 2
- 5 10 15 4 8 12 3 6 9 2 4 6 1 2 3
- 1 2 2 4 3 6 4 8 5 10

18

Consider the following method.

```
public void sample(int num1, int num2) {
      int result = 99;

if (num1==num2) {result = 0;}
else if (num1>num2){result = 1;}
else {result = -1;}
System.out.println(result);
}
```

Which of the following methods will print the same values (0,1,-1) as the method above no matter what values are passed for `num1` and `num2`?

I.

```
public void method1(int num1, int num2) {
      int result=99;

      if (num1 == num2) {result = 0;}
      else {
         if(num1 > num2) {result = 1;}
         else {result = -1;}
      }
System.out.println(result);
}
```

II.

```
public void method2(int num1, int num2) {
      int result = 99;

      if (num1 == num2) {result = 0;}
      if (num1 >= num2) {result = 1;}
      else {result = -1;}
System.out.println(result);
}
```

III.

```
public void method3(int num1, int num2) {
      int result = 99 ;

      if (num1 == num2) {result = 0;}
      if (num1 > num2) {result = 1;}
      if (num1 < num2) {result = -1;}
      System.out.println(result);
}
```

- I and III
- I only
- II only
- II and III
- I, II, and III

19

What are the first and last values output by the following code segment?

```
int t = 13;
while (t < 29)
{
   System.out.println(t);
   t++;
}
```

First	Last
13	28
13	29
14	28
14	29
1	28

20

Given the following code:

```
String s1 = new String("hi");
String s2 = new String("hi");
String s3 = s2;
```

Which of the following would return true:

```
I.  s1.equals(s2)
II. s1 == s2
III. s2.equals(s3);
IV. s2 == s3;
```

- I and III
- All are true
- I, III, and IV
- II and IV
- III and IV

21

Consider the following partial class definitions.

```
public class C1 {
  private int num;
  private String name;

  public C1(int theNum) {
    num = theNum;
  }

  public C1(String theName) {
    name = theName;
  }
  // other methods not shown
}

public class C2 extends C1 {
   // methods not shown
}
```

Which of the following constructors are valid for C2?

```
I.  public C2 () { }
II. public C2 (int quan) {super (quan); }
III. public C2 (String label) { super(label); }
```

- All three are valid
- II only
- III only
- II and III
- None are valid

22

The Boolean expression (x==y && !(x==y)) || (x!=y && !(x!=y)) can be simplified to which of the following?

- x != y
- x == y
- true
- false
- x < y

23

```
public static void sort(int[] a) {
    int maxCompare = a.length - 1;
    int savedIndex = 0;
    int numSteps = 0;
    int temp = 0;

    for (int i = maxCompare; i > 0; i--) {
      savedIndex = i;

            for (int j = i - 1; j >= 0; j--) {
              /* missing code */
       }

      temp = a[i];
      a[i] = a[savedIndex];
      a[savedIndex] = temp;
    }
  }
```

Which of the following could be used to replace `/* missing code */` so that it sorts the array `a` in ascending order?

- `if (a[savedIndex > a[j]) { j = savedIndex; }`
- `if (a[j] > a[savedIndex]) { savedIndex = j;}`
- `if (a[j] < a[savedIndex]) { savedIndex = j; }`
- `if (a[j] > a[savedIndex]) { j = savedIndex;}`
- `if (a[j] == a[savedIndex]) { savedIndex = j; }`

24

Which of the following statements about interfaces is (are) true?

I. One interface can inherit from another

II. All methods declared in an interface are abstract methods (can't have a method body).

III. All methods declared in an interface are public methods.

- II only
- III only
- I and II only
- I, II, and III
- I only

25

A two-dimensional array is used to represent a matrix. The declaration is below:

```
int[][] matrix = new int[2][3];
```

Consider the following method:

```
public static void changeMatrix(int[][] matrix )
{
       for (int y = 0; y < matrix.length;y++)
          for(int x = 0; x < matrix[y].length; x++)
             if(y==x)
                matrix[y][x] = Math.abs(matrix[y][x]);
}
```

If matrix is initialized to be: {-1, -2, 3},{4, -5, 6}. What will the values in matrix be after changeMatrix(matrix) is called?

- {4, -5, 6},{-1, -2, 3}
- {4, 5, 6},{1, 2, 3}
- {1, 2, 3},{4, 5, 6}
- {-1, -2, 3},{4, -5, 6}
- {1, -2, 3},{4, 5, 6}

26

What are the values of `a` and `b` after the for loop finishes?

```
int a = 5, b = 2, temp;
for (int i=1; i<=4; i++) {
   temp = a;
   a = i + b;
   b = temp - i;
}
```

- a = 4 and b = 3
- a = 7 and b = 0
- a = 2 and b = -2
- a = 5 and b = 2
- a = 9 and b = 2

27

Condsider the following method. What value is returned from a call of mystery(4)?

```
public static int mystery(int n)
{
  if (n == 0)
    return 1;
  else
    return 3 * mystery (n - 1);
}
```

- 243
- 0
- 3
- 81
- 27

28

Which of the following correctly shows the iterations of an ascending (from left to right) insertion sort on an array with the following elements: {6,3,8,5,1}?

- {3,6,8,5,1}, {3,5,6,8,1}, {1,3,5,6,8}
- {1,3,8,5,6}, {1,3,8,5,6}, {1,3,5,8,6}, {1,3,5,6,8}
- {3,6,8,5,1}, {3,6,8,5,1}, {3,5,6,8,1}, {1,3,5,6,8}
- {1,3,8,5,6}, {1,3,5,8,6}, {1,3,5,6,8}
- {1,6,3,8,5}, {1,3,6,8,5}, {1,3,5,6,8}

29

Consider the following code segment

```
for(int i = 0; i < 3; i++) {
    for(int j = 1; j <= 7; j++)
        System.out.println("*");
}
```

How many times will a '*' be printed?

- 21
- 18
- 32
- 28
- 16

30

Consider the following method.

```
public static void conditionTest(int num1, int num2)
{
  if ((num1>0) && (num2>0)) {
     if (num1>num2)
       System.out.println("A");
     else
       System.out.println("B");
  }
  else if ((num2<0) && (num1<0)) {
     System.out.println("C");
  }
  else if (num2 < 0) {
     System.out.println("D");
  }
  else {
     System.out.println("E");
  }
}
```

What is the output from `conditionTest(-3,2)`?

- A
- B
- C
- D
- E

31

Consider the following:

```
String s1 = "Hi There";
String s2 = s1;
String s3 = s2;
String s4 = s1;
s2 = s2.toUpperCase();
s3 = s3.toLowerCase();
s4 = null;
```

What is value of s1 after the above code executes?

- hi there
- HI THERE
- Hi There
- null
- hI tHERE

32

What is the output from the following code?

```
String s = "Computer Science is fun!";
String s1 = s.substring(0,8);
String s2 = s1.substring(2);
String s3 = s2.substring(0,3);
System.out.println(s3);
```

- mp
- mpu
- mpur
- omp
- om

33

Given the following partial class definitions:

```
public class Book implements Comparable
{ // code for class
}
public class Dictionary extends Book
{ // code for class
}
```

Which declaration will result in a compiler error?

- Book b = new Book();
- Dictionary d = new Book();
- Comparable c = new Book();
- Book b = new Dictionary ();
- Comparable c = new Dictionary();

34

Given the following code:

```
public static int mystery(String str)
{
  if (str.length() == 0) return 0;
  else
  {
    if (str.substring(0,1).equals("x")) return 1 +
                                mystery(str.substring(1));
    else return mystery(str.substring(1));
  }
}
```

What will it return when called with `mystery("xxzxyxx")`?

- 2
- 5
- 1
- 4
- 0

35

Which will cause the **longest** execution of a binary search looking for a value in an array of 10 integers?

- The value is the first one in the array
- The value is in the middle of the array
- The value is at position 3 in the array
- The value isn't in the array
- The value is at position 6 in the array

36

If you have a parent class Animal that has a method speak() which returns "Awk" and you have children classes that do the following:

```
Cat has a speak method that returns "Meow"
Bird doesn't have a speak method
Dog has a speak method that returns "Woof"
Pig doesn't have a speak method
Cow has a speak method that returns "Moo"
```

What is the output from looping through this array of animals and asking each to speak()?

```
Animal[] a = { new Cat(), new Cow(), new Dog(), new Pig(), new Bird() }
```

- Awk Awk Awk Awk Awk
- This won't compile
- This will have runtime errors
- Meow Moo Woof Oink Awk
- Meow Moo Woof Awk Awk

37

What is the result of 17_{16} - 13_{8}?

- 4_{8}
- 4_{16}
- 00001100_{2}
- 00000010_{2}
- 4_{10}

38

Consider the following code segment

```
public static void test(int[] a, int y)
{
  if (a.length > 1)
      a[1] = a[1] * 2;
  y = y * 2;
}
```

What are the values of `s` and `b` after the following has executed?

```
int[] s = {3,4};
int b = 4;
test(s,b);
```

- `s={3, 8}; b=4;`
- `s={3, 4}; b=4;`
- `s={6, 4}; b=4;`
- `s={3, 8}; b=8;`
- `s={6, 8}; b=8;`

39

Which of the following is (are) true?

I. Insertion sort takes longer when the array is sorted in ascending order and you want it sorted in descending order.

II. Merge sort uses recursion.

III. Selection sort takes less time to execute if the array is already sorted in the correct order.

- I only
- II only
- III only
- I and II only
- I , II, and III

40

Give the following code:

```
private int[] arr;
public int mystery(int low, int high, int num) {
  int mid = (low+high) / 2;
  if (low > high) {
    return -1;
  }
  else if (arr[mid] < num) {
    return mystery(mid +1, high, num);
  }
  else if (arr[mid] > num) {
    return mystery(low, mid - 1, num);
  }
  else
    return mid;
}
```

How many calls to mystery are made (including the first call) of `mystery(0, 4, 5)` when `arr` = {1, 2, 3, 5, 7}?

- 1
- 2
- 3
- 4
- 5