

7. Consider the following loop, where n is some positive integer.

```
for (int i = 0; i < n; i += 2)
{
    if (/* test */)
        /* perform some action */
}
```

In terms of n , which Java expression represents the maximum number of times that */* perform some action */* could be executed?

- (A) $n / 2$
- (B) $(n + 1) / 2$
- (C) n
- (D) $n - 1$
- (E) $(n - 1) / 2$

Loops &
ARRAYS
packet #1

Refer to the following class for Questions 13 and 14.

```
public class Tester
{
    private int[] testArray = {3, 4, 5};

    /** @param n an int to be incremented by 1 */
    public void increment (int n)
    { n++; }

    public void firstTestMethod()
    {
        for (int i = 0; i < testArray.length; i++)
        {
            increment(testArray[i]);
            System.out.print(testArray[i] + " ");
        }
    }

    public void secondTestMethod()
    {
        for (int element : testArray)
        {
            increment(element);
            System.out.print(element + " ");
        }
    }
}
```

13. What output will be produced by invoking `firstTestMethod` for a `Tester` object?
- (A) 3 4 5
 - (B) 4 5 6
 - (C) 5 6 7
 - (D) 0 0 0
 - (E) No output will be produced. An `ArrayIndexOutOfBoundsException` will be thrown.
14. What output will be produced by invoking `secondTestMethod` for a `Tester` object, assuming that `testArray` contains 3, 4, 5?
- (A) 3 4 5
 - (B) 4 5 6
 - (C) 5 6 7
 - (D) 0 0 0
 - (E) No output will be produced. An `ArrayIndexOutOfBoundsException` will be thrown.

Questions 27 and 28 refer to the search method in the Searcher class below.

```
public class Searcher
{
    private int[] arr;

    /** Constructor. Initializes arr with integers. */
    public Searcher()
    { /* implementation not shown */ }

    /** Precondition: arr[first]...arr[last] sorted in ascending order.
     * Postcondition: Returns index of key in arr. If key not in arr,
     *                 returns -1.
     */
    public int search(int first, int last, int key)
    {
        int mid;
        while (first <= last)
        {
            mid = (first + last) / 2;
            if (arr[mid] == key) //found key, exit search
                return mid;
            else if (arr[mid] < key) //key to right of arr[mid]
                first = mid + 1;
            else //key to left of arr[mid]
                last = mid - 1;
        }
        return -1; //key not in list
    }
}
```

27. Which assertion is true just before each execution of the while loop?

- (A) $arr[first] < key < arr[last]$
- (B) $arr[first] \leq key \leq arr[last]$
- (C) $arr[first] < key < arr[last]$ or key is not in arr
- (D) $arr[first] \leq key \leq arr[last]$ or key is not in arr
- (E) $key \leq arr[first]$ or $key \geq arr[last]$ or key is not in arr

28. Consider the array a with values as shown:

4, 7, 19, 25, 36, 37, 50, 100, 101, 205, 220, 271, 306, 321

where 4 is a[0] and 321 is a[13]. Suppose that the search method is called with $first = 0$ and $last = 13$ to locate the key 205. How many iterations of the while loop must be made in order to locate it?

- (A) 3
- (B) 4
- (C) 5
- (D) 10
- (E) 13

Questions 6 and 7 refer to the Deck class described below.

A Deck class contains an array `cards` with an even number of Card values and a final variable `NUMCARDS`, which is an odd integer.

6. Here are two possible algorithms for shuffling the deck.

Algorithm 1

Initialize an array of Card called `shuffled` of length `NUMCARDS`.

Set `k` to 0.

For `j=0` to `NUMCARDS/2-1`

- Copy `cards[j]` to `shuffled[k]`

- Set `k` to `k+2`

Set `k` to 1.

For `j=NUMCARDS/2` to `NUMCARDS-1`

- Copy `cards[j]` to `shuffled[k]`

- Set `k` to `k+2`

Algorithm 2

Initialize an array of Card called `shuffled` containing `NUMCARDS` slots.

For `k=0` to `NUMCARDS-1`

- Repeatedly generate a random integer `j` from 0 to `NUMCARDS-1`,
until `cards[j]` contains a card not marked as empty

- Copy `cards[j]` to `shuffled[k]`

- Set `cards[j]` to empty

Which is a *false* statement concerning Algorithms 1 and 2?

- (A) A disadvantage of Algorithm 1 is that it won't generate all possible deck permutations.
- (B) For Algorithm 2, to determine the last element shuffled requires an average of `NUMCARDS` calls to the random number generator.
- (C) Algorithm 2 will lead to more permutations of the deck than Algorithm 1.
- (D) In terms of run time, Algorithm 2 is more efficient than Algorithm 1.
- (E) If Algorithm 1 is repeated several times, it may return the deck to its original state.

7. The following shuffle method is used to shuffle the cards in the Deck class.

```
Line 1: public void shuffle()
Line 2: {
Line 3:     for (int k = NUMCARDS; k > 0; k--)
Line 4:     {
Line 5:         int randPos = (int) (Math.random() * (k + 1));
Line 6:         //swap randomly selected card with card at position k
Line 7:         Card temp = cards[k];
Line 8:         cards[k] = cards[randPos];
Line 9:         cards[randPos] = temp;
Line 10:     }
Line 11: }
```

The method does not work as intended. Which of the following changes should be made to correct the method?

- (A) Replace Line 3 with
for (int k = NUMCARDS; k >= 0; k--)
- (B) Replace Line 3 with
for (int k = NUMCARDS - 1; k > 0; k--)
- (C) Replace Line 3 with
for (int k = 1; k <= NUMCARDS; k++)
- (D) Replace Line 5 with
int randPos = (int) (Math.random() * k);
- (E) Replace Lines 7 - 9 with
Card temp = cards[randPos];
cards[randPos] = cards[k];
cards[k] = temp;

5. Consider this program segment. You may assume that wordList has been declared as ArrayList<String>.

```
for (String s : wordList)
    if (s.length() < 4)
        System.out.println("SHORT WORD");
```

What is the maximum number of times that SHORT WORD can be printed?

- (A) 3
(B) 4
(C) wordList.size()
(D) wordList.size() - 1
(E) s.length()

12. The following shuffling method is used to shuffle an array `arr` of `int` values. The method assumes the existence of a `swap` method, where `swap(arr, i, j)` interchanges the elements `arr[i]` and `arr[j]`.

```
public static void shuffle (int[] arr)
{
    for (int k = arr.length - 1; k > 0; k--)
    {
        int randIndex = (int) (Math.random() * (k + 1));
        swap(arr, k, randIndex);
    }
}
```

Suppose the initial state of `arr` is 1 2 3 4 5, and when the method is executed the values generated for `randIndex` are 3, 2, 0, and 1, in that order. What will be the final state of `arr`?

- (A) 5 2 1 3 4
 - (B) 1 2 5 3 4
 - (C) 5 4 1 3 2
 - (D) 4 5 1 3 2
 - (E) 2 5 1 3 4
19. Consider method `getCount` below:

```
public static int getCount(String s, String sub)
{
    int count = 0;
    int pos = s.indexOf(sub);
    while (pos >= 0)
    {
        s = s.substring(pos);
        count++;
        pos = s.indexOf(sub);
    }
    return count;
}
```

What will the method call `getCount("a carrot and car", "car")` return?

- (A) 0
- (B) 1
- (C) 2
- (D) 3
- (E) No value returned. The method is in an infinite loop.

13. Refer to

```
/**
 *
 *
 *
 */
pub
{
}
The
re
For
(A)
(B)
(C)
(D)
(E)
```

14. A
ha
ho
re
to
si
n

21. Consider the following program segment:

```
/** Precondition: a[0]...a[n-1] is an initialized array of integers,  
 * and 0 < n <= a.length.  
 */  
int c = 0;  
for (int i = 0; i < n; i++)  
    if (a[i] >= 0)  
    {  
        a[c] = a[i];  
        c++;  
    }  
n = c;
```

Which is the best postcondition for the segment?

- (A) $a[0] \dots a[n-1]$ has been stripped of all positive integers.
- (B) $a[0] \dots a[n-1]$ has been stripped of all negative integers.
- (C) $a[0] \dots a[n-1]$ has been stripped of all nonnegative integers.
- (D) $a[0] \dots a[n-1]$ has been stripped of all occurrences of zero.
- (E) The updated value of n is less than or equal to the value of n before execution of the segment.

10. Refer to the following program segment.

```
for (int n = 50; n > 0; n = n / 2)  
    System.out.println(n);
```

How many lines of output will this segment produce?

- (A) 50
- (B) 49
- (C) 7
- (D) 6
- (E) 5

12. Consider the following static method.

```
public static int compute(int n)  
{  
    for (int i = 1; i < 4; i++)  
        n *= n;  
    return n;  
}
```

Which of the following could replace the body of `compute`, so that the new version returns the identical result as the original for all n ?

- (A) `return 4 * n;`
- (B) `return 8 * n;`
- (C) `return 64 * n;`
- (D) `return (int) Math.pow(n, 4);`
- (E) `return (int) Math.pow(n, 8);`

15. Consider the following code segment.

```
int num1 = value1, num2 = value2, num3 = value3;
while (num1 > num2 || num1 > num3)
{
    /* body of loop */
}
```

You may assume that `value1`, `value2`, and `value3` are `int` values. Which of the following is sufficient to guarantee that `/* body of loop */` will never be executed?

- (A) There is no statement in `/* body of loop */` that leads to termination
- (B) `num1 < num2`
- (C) `num1 < num3`
- (D) `num1 > num2 && num1 > num3`
- (E) `num1 < num2 && num1 < num3`

16. Cons

pu
{

1) Consider the following code segment.

```
int a = 0;
int i = 4;

while (a < 4)
{
    a = a + 4;
    System.out.println(a + " ");
}
```

What is the output of above code segment?

- (A) Nothing, the loop terminates at start
- (B) 4 5
- (C) 4
- (D) 0 4
- (E) 4 4

3) Consider the following code segment.

```
String l = "zeshan";

for (int i = l.length() - 1; i >= 0; i--)
{
    System.out.print(l.charAt[i]);
}
```

What is the output of above code segment?

- (A) shanze
- (B) nshzea
- (C) hzeasn
- (D) nahsez
- (E) None of the above

13) Consider the following code segment.

```
int x = 24;
int y = 30;
while (y != 30)
{
    int z = x % y;
    x = y;
    y = z;
}

System.out.println(x);
```

What is the output of above code segment?

- (A) 0
- (B) 6
- (C) 12
- (D) 24
- (E) 30

26) Consider the following code segment.

```
for (int k = 0; k < 20; k = k + 2)
{
    if (k % 3 == 1)
    {
        System.out.print(k + " ");
    }
}
```

For Questions 30 - 31, refers to the following code.

```
int y = 1;
int x = 123;
while (x > 0)
{
    y *= 10;
    y += x % 10;
    x /= 10;
}
```

What will be displayed upon execution of the code?

(A) 4 10 16

(B) 4 5 6 7 2 4 5

(C) 3

(D) 5 64 22 33

(E) None of the above

30) What will be printed when we execute the following code segment?

```
System.out.println(y);
```

(A) 1000

(B) 1320

(C) 1321

(D) 1329

(E) None of the above

31) What will be printed when we execute the following code segment?

```
System.out.println(x);
```

(A) 0

(B) 1

(C) 2

(D) 3

(E) 4

35) Consider the following code segment.

```
int x = 2005;

for (int j = 0; j < 50; j++)
{
    x = (x + 3) / 2;
}
```

What will the value of x be after the code has been executed?

- (A) 0
- (B) 1
- (C) 2
- (D) 3
- (E) 65

38) Consider the following code segment.

```
1 int x = 0;
2 while (1)
3 {
4     System.out.print("x plus one is " + (x + 1));
5 }
```

Which of the following statements is true?

- I. Syntax error in Line 1
- II. Syntax error in Line 2
- III. An infinite loop will occur

- (A) I only
- (B) II only
- (C) I and II
- (D) I and III
- (E) III only

1) Consider the following code segment.

```
while (0 > 4)
{
    System.out.println("You are hacked");
}
```

What is the output of the above code segment?

I. Compile time error

II. You are hacked

III. Loop prints infinitely

(A) I only

(B) II only

(C) III only

(D) II and III

(E) None

2) Consider the following code segment.

```
1 int i = 0;
2 while (i < 4)
3 {
4     System.out.println("Infinite loop");
5 }
```

How many times is Line 4 executed?

(A) 4

(B) Infinite times

(C) 0

(D) 5

(E) 3

15) What will be displayed upon execution of this code segment?

```
int [] a = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
for (int i = 0; i < a.length; i++)
{
    if (a[i] % 2 == 0)
    {
        System.out.print(a[i] + " ");
    }
}
```

(A) 1 3 4 5 6 7

(B) 2 4 6 8 10

(C) 1 2 3 4 5

(D) 2 4 5 6 7

(E) None of the above

18) Consider the following code segment.

```
int m = 20;
int n = 2;
int temp;

for (int count = 1; count <= 20; count++)
{
    temp = m;
    m = n + count;
    n = temp - count;
}
```

What are the values of m and n after executing the code?

(A) m: 20 n: -8

(B) m: 9 n: -9

(C) m: 30 n: -8

(D) m: 29 n: -6

(E) m: 4 n: -5

21) Consider the following method.

```
1 public void printer(int i)
2 {
3     while (i < 10)
4     {
5         System.out.println("Printing.....");
6         i++;
7     }
8 }
```

What should the value of *i* be in the printer() method so that Line 5 is executed 10 times?

- (A) 5
- (B) 0
- (C) 10
- (D) 15
- (E) 7

22) Consider the following code segment.

```
1 int count = 0;
2 while (count++ < 10)
3 {
4     System.out.println("Line " + count);
5 }
```

How many times is Line 4 executed?

- (A) 5
- (B) 6
- (C) 7
- (D) 9
- (E) 10

For Questions 23 - 24, refer to the following code.

```
public class Employee
{
    private String EmpId;
    private String Name;
    private double Salary;

    public Employee(String EmpId, String Name, double Salary)
    {
        this.EmpId = EmpId;
        this.Name = Name;
        this.Salary = Salary;
    }
    // Setter and Getter methods
}

Employee[] emp = new Employee[50];
```

23) Which of the following code segments can be used to print the maximum salary?

```
(A) for (int i = 0; i < emp.length; i++)
{
    int max = 0;
    if (emp.getSalary() > max)
    {
        max = emp.getSalary();
    }
}
System.out.println(max);
```

```
(B) int max = Employee[0];
    for (Employee e : emp)
    {
        if (e.getSalary() > max)
        {
            max = e.getSalary();
        }
    }
    System.out.println(max);
```

```
(C) if(emp[i].getSalary > emp[i + 1].getSalary())
    {
        max = emp[i].getSalary();
    }
    System.out.println(max);
```

```
(D) for (int i = 0; i > emp.length; i--)
    {
        int max = 0;
        if (emp.getSalary() < max)
        {
            max = emp.getSalary();
        }
    }
    System.out.println(max);
```

(E) None of the above

24) Which of the following code segments represents the total salary of all the employees?

```
(A) double total_sal = 0.0;
    for (Employee e : Emp)
    {
        total_sal = e.getSalary();
    }
```

```
(B) double total_sal = 0.0;
    for (Employee e : Emp)
    {
        total_sal = e[i].getSalary();
    }
```

```
(C) double total_sal = 0.0;
    for (Employee e : Emp)
    {
        total_sal += e.getSalary();
    }
```

```
(D) double total_sal = 0.0;
    for (Employee e : Emp)
    {
        total_sal += Emp.getSalary();
    }
```

(E) None of the above

5) Consider the following code segment.

```
for (int i = 0; i <= 1; i++)
{
    for (int j = 0; j < 2; j++)
    {
        if (i == j)
        {
            // nothing happens
        }
        else
        {
            System.out.println("i = " + i + " , j = " + j);
        }
    }
}
```

What will be displayed upon execution of this code segment?

- (A) i = 0, j = 1
i = 1, j = 0
- (B) i = 0, j = 1
i = 0, j = 2
- (C) i = 0, j = 0
i = 1, j = 1
- (D) i = 1, j = 2

6) Consider the following code segment.

```
for (int i = 2; i > 10; i--)  
{  
    System.out.print("i = " + i + " ");  
}
```

What will be displayed upon execution of this code segment?

(A) The statements will not compile.

(B) i = 3 i = 2

(C) i = 2 i = 1

(D) i = 1

(E) Nothing will be displayed

7) Consider the following code segment.

```
for (int x = 5; x <= 50; x += 5)
{
    System.out.print(" " + x);
}
```

Which of the following while loops are equivalent to the for loop above?

(A)

```
int x = 5;
while (x <= 50)
{
    System.out.print(" " + x);
    x += 5;
}
```

(B)

```
while (x <= 50)
{
    int x = 5;
    System.out.print(" " + x);
    x += 5;
}
```

(C)

```
while (int x = 5; x <= 50)
{
    System.out.print(" " + x);
    x += 5;
}
```

(D)

```
int x = 5;
while (x <= 50; x += 5)
{
    System.out.print(" " + x);
}
```

9) Which of the following for loops can be used to iterate from 1 to 10?

(A)

```
for (i <= 10; i++)
{
    System.out.println(i);
}
```

(B)

```
for (int i = 1; i <= 10; i++)
{
    System.out.println(i);
}
```

(C)

```
for (int i = 1; i < 10; i++)
{
    System.out.println(i);
}
```

(D)

```
for (int i = 1; i <= 10; i--)
{
    System.out.println(i);
}
```

8) Consider the following code segment.

```
int x = 0;
while (x <= 3)
{
    System.out.print(" " + x);
    x++;
}
```

(A) The statements will not compile.

(B) Nothing will be displayed.

(C) 0 1 2

(D) 0 1 2 3

(E) 0 1 2 3 4

11) Which of the following while loops can be used to iterate from 1 to 7?

(A)

```
int i = 1;
while (i < 7)
{
    System.out.println(i);
    i++;
}
```

(B)

```
int i = 1;
while (i <= 7)
{
    System.out.println(i);
    i++;
}
```

(C)

```
int i = 1;
while (i <= 7)
{
    System.out.println(i);
}
```

(D)

```
int i = 1;
while (i = 1; i < 7)
{
    System.out.println(i);
    i++;
}
```

10) Consider the following code segment.

```
int i;
for (i = 1; i <= 10; i++)
{
    System.out.println(i);
}

int result = i;
System.out.println(result);
```

What will be displayed upon execution of this code segment?

(A) 10

(B) 11

(C) 9

(D) The code will not compile.

12) Consider the following code segment.

```
int i = 1;
while (i < 10)
{
    System.out.print(i);
}
```

What will be displayed upon execution of this code segment?

- (A) 123456789
- (B) 1
- (C) There will be an endless loop.
- (D) 12345678910

13) Consider the following code segment.

```
for (int i = 0; i >= 0; i++)
{
    System.out.println("a");
}
```

What will be displayed upon execution of this code segment?

- (A) a
- (B) The code will not compile.
- (C) There will be an endless loop.
- (D) Nothing will be displayed.

6. Refer to the following method that finds the smallest value in an array.

```
/** Precondition: arr is initialized with int values.
 * @param arr the array to be processed
 * @return the smallest value in arr
 */
public static int findMin(int[] arr)
{
    int min = /* some value */;
    int index = 0;
    while (index < arr.length)
    {
        if (arr[index] < min)
            min = arr[index];
        index++;
    }
    return min;
}
```

Which replacement(s) for */* some value */* will always result in correct execution of the `findMin` method?

- I Integer.MIN_VALUE
- II Integer.MAX_VALUE
- III arr[0]

- (A) I only
- (B) II only
- (C) III only
- (D) I and III only
- (E) II and III only

GO ON TO THE NEXT PAGE.

29. Consider the following RandomList class.

```
public class RandomList
{
    private int[] ranList;

    public RandomList()
    { ranList = getList(); }

    /** @return array with random Integers from 0 to 100
     *   inclusive */
    public int[] getList()
    {
        System.out.println("How many integers? ");
        int listLength = IO.readInt(); //read user input
        int[] list = new int[listLength];
        for (int i = 0; i < listLength; i++)
        {
            /* code to add integer to list */
        }
        return list;
    }

    /** Print all elements of this list. */
    public void printList()
    { ...
}
}
```

Which represents correct */* code to add integer to list */*?

- (A) list[i] = (int) (Math.random() * 101);
- (B) list.add((int) (Math.random() * 101));
- (C) list[i] = (int) (Math.random() * 100);
- (D) list.add(new Integer(Math.random() * 100))
- (E) list[i] = (int) (Math.random() * 100) + 1;

8. A method is to be written to search an array for a value that is larger than a given item and return its index. The problem specification does not indicate what should be returned if there are several such values in the array. Which of the following actions would be best?
- (A) The method should be written on the assumption that there is only one value in the array that is larger than the given item.
 - (B) The method should be written so as to return the index of every occurrence of a larger value.
 - (C) The specification should be modified to indicate what should be done if there is more than one index of larger values.
 - (D) The method should be written to output a message if more than one larger value is found.
 - (E) The method should be written to delete all subsequent larger items after a suitable index is returned.

35. Consider method `findSomething` below:

```
/** Precondition: a.length is equal to b.length. */
public static boolean findSomething(int[] a, int[] b)
{
    for (int aValue: a)
    {
        boolean found = false;
        for (int bValue: b)
        {
            if (bValue == aValue)
                found = true;
        }
        if (!found)
            return false;
    }
    return true;
}
```

Which *best* describes what method `findSomething` does? Method `findSomething` returns true only if

- (A) Arrays `a` and `b` contain identical elements in the same order.
- (B) Arrays `a` and `b` contain identical elements in reverse order.
- (C) Arrays `a` and `b` are permutations of each other.
- (D) Array `a` contains at least one element that is also in `b`.
- (E) Every element of array `a` is also in `b`.

40. Consider a method `partialProd` that returns an integer array `prod` such that for all k , `prod[k]` is equal to $arr[0] * arr[1] * \dots * arr[k]$. For example, if `arr` contains the values `{2,5,3,4,10}`, the array `prod` will contain the values `{2,10,30,120,1200}`.

```
public static int[] partialProd(int[] arr)
{
    int[] prod = new int[arr.length];
    for (int j = 0; j < arr.length; j++)
        prod[j] = 1;
    /* missing code */
    return prod;
}
```

Consider the following two implementations of `/* missing code */`.

Implementation 1

```
for (int j = 1; j < arr.length; j++)
{
    prod[j] = prod[j - 1] * arr[j];
}
```

Implementation 2

```
for (int j = 0; j < arr.length; j++)
    for (int k = 0; k <= j; k++)
    {
        prod[j] = prod[j] * arr[k];
    }
```

Which of the following statements is *true*?

- (A) Both implementations work as intended but Implementation 1 is faster than Implementation 2.
- (B) Both implementations work as intended but Implementation 2 is faster than Implementation 1.
- (C) Both implementations work as intended and are equally fast.
- (D) Implementation 1 doesn't work as intended because the elements of `prod` are incorrectly assigned.
- (E) Implementation 2 doesn't work as intended because the elements of `prod` are incorrectly assigned.

11. Consider the following method.

```
public static void alterArray(int[] arr)
{
    int mid = arr.length/2;
    for (int i = 0; i < mid; i++)
    {
        int temp = arr[i];
        arr[i] = arr[arr.length - i - 1];
        arr[arr.length - i - 1] = temp;
    }
}
```

If the current state of a matrix mat is

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

which matrix will result from the method call `alterArray(mat[2])`?

- (A) 2 7 9 5
3 4 1 8
6 5 0 9
- (B) 2 7 0 5
8 1 4 3
6 5 9 9
- (C) 5 9 7 2
3 4 1 8
9 0 5 6
- (D) 2 7 9 5
8 1 4 3
9 0 5 6
- (E) 5 9 7 2
8 1 4 3
6 5 0 9

12. Consider the following:

You
Lanc
decl
I
II
(A
(E
(C
(I
(I

14. A `Clock` class has `hours`, `minutes`, and `seconds` represented by `int` values. It also has each of the following methods: `setTime` to change the time on a `Clock` to the hour, minute, and second specified; `getTime` to access the time; and `toString` to return the time as a `String`. The `Clock` class has a constructor that allows a `Clock` to be created with three `int` parameters for hours, minutes, and seconds. Consider a two-dimensional array of `Clock` values called `allClocks`. A code segment manipulating `allClocks` is as follows:

```
for (Clock[] row : allClocks)
    for (Clock c : row)
        /* more code */
```

Assuming the `Clock` class works as specified, which replacement for `/* more code */` will cause an error?

- I `System.out.print(c);`
 - II `c.setTime(0, 0, 0);`
 - III `c = new Clock(0, 0, 0);`
- (A) I only
(B) II only
(C) III only
(D) II and III only
(E) I and II only

5. The following shuffle algorithm is used to shuffle an array of `int` values, `nums`.

```
public void shuffle ()
{
    for (int k = nums.length - 1; k > 0; k--)
    {
        int randPos = (int) (Math.random() * (k + 1));
        int temp = nums[k];
        nums[k] = nums[randPos];
        nums[randPos] = temp;
    }
}
```

Suppose the initial state of `nums` is 8, 7, 6, 5, 4, and when the method is executed the values generated for `randPos` are 3, 2, 0, 0, in that order. What element will be contained in `nums[2]` after execution?

- (A) 8
(B) 7
(C) 6
(D) 5
(E) 4

13. Consider the following instance variable and method:

```
private int[] nums;

/** Precondition: nums contains int values in no particular order.
 */
public int getValue()
{
    for (int k = 0; k < nums.length; k++)
    {
        if (nums[k] % 2 != 0)
            return k;
    }
    return -1;
}
```

Suppose the following statement is executed:

```
int j = getValue();
```

If the value returned in `j` is a positive integer, which of the following best describes the contents of `nums`?

- (A) The only odd int in `nums` is at position `j`.
 - (B) All values in positions 0 through `j-1` are odd.
 - (C) All values in positions 0 through `j-1` are even.
 - (D) All values in positions `nums.length-1` down to `j+1` are odd.
 - (E) All values in positions `nums.length-1` down to `j+1` are even.
22. A class of 30 students rated their computer science teacher on a scale of 1 to 10 (1 means awful and 10 means outstanding). The `responses` array is a 30-element integer array of the student responses. An 11-element array `freq` will count the number of occurrences of each response. For example, `freq[6]` will count the number of students who responded 6. The quantity `freq[0]` will not be used. Here is a program that counts the students' responses and outputs the results.

```
public class StudentEvaluations
{
    public static void main(String args[])
    {
        int[] responses = {6,6,7,8,10,1,5,4,6,7,
                          5,4,3,4,4,9,8,6,7,10,
                          6,7,8,8,9,6,7,8,9,2};
        int[] freq = new int[11];
        for (int i = 0; i < responses.length; i++)
            freq[responses[i]]++;
        //output results
        System.out.print("rating" + " " + "frequency\n");
        for (int rating = 1; rating < freq.length; rating++)
            System.out.print(rating + " " +
                             freq[rating] + "\n");
    }
}
```

Suppose the last entry in the initializer list for the `responses` array was incorrectly typed as 12 instead of 2. What would be the result of running the program?

- (A) A rating of 12 would be listed with a frequency of 1 in the output table.
- (B) A rating of 1 would be listed with a frequency of 12 in the output table.
- (C) An `ArrayIndexOutOfBoundsException` would be thrown.
- (D) A `StringIndexOutOfBoundsException` would be thrown.
- (E) A `NullPointerException` would be thrown.

26. Consider an array `arr` that is initialized with `int` values. The following code segment stores in `count` the number of positive values in `arr`.

```
int count = 0, index = 0;
while (index < arr.length)
{
    if (arr[index] > 0)
        count++;
    index++;
}
```

Which of the following is equivalent to the above segment?

```
I int count = 0;
   for (int num : arr)
   {
       if (arr[num] > 0)
           count++;
   }
```

```
II int count = 0;
    for (int num : arr)
    {
        if (num > 0)
            count++;
    }
```

```
III int count = 0;
     for (int i = 0; i < arr.length; i++)
     {
         if (arr[i] > 0)
             count++;
     }
```

- (A) I only
(B) II only
(C) III only
(D) II and III only
(E) I and III only

- 4) Consider the incomplete method below. The method `index()` is intended to return the index of number passed to it in the array. If the number is not found, the method returns -1.

```
public int index(int x)
{
    int[] array={1,2,3,5};

    // missing code
}
```

Which of the code segments shown below can be used to replace `// missing code` so that `index()` will work as intended?

27. A squa

int
where
put
{

As
m
(
(

28.

```
(A) for (int i = 0; i < 4; i++)
    {
        if (x == array[i])
            {
                return i;
            }
    }
return -1;
```

```
(B) for (int i = 0; i < 4; i++)
    {
        if (x == array[i])
            {
                return i;
            }
        else
            {
                return i;
            }
    }
return -1;
```

```
(C) for (int i = 0; i < 4; i++)
    {
        if (x == array[i])
            {
                return i;
            }
    }
}
```

```
(D) for (int i = 0; i < 4; i++)
    {
        if (x == array[i])
            {
                return i;
            }
        else
            {
                return i;
            }
    }
}
```

(E) None of the above

For Questions 32 - 34, refer to the following code.

```
public static void divider(int[] a, int[] q, int[] r)
{
    q[0] = a[0] / 5;
    r[0] = a[0] % 5;
}
```

32) What will be printed when we execute the following code segment?

```
int[] x = {21, 22, 23};
int[] y = new int[3];
int[] z = {21, 22, 23};

divider(x, y, z);
System.out.println(x[0]);
```

- (A) 41
- (B) 31
- (C) 21
- (D) 0
- (E) 33

33) What will be printed when we execute the following code segment?

```
int[] x = {21, 22, 23};  
int[] y = new int[3];  
int[] z = {21, 22, 23};  
  
divider(x, y, z);  
System.out.println(y[1]);
```

- (A) 0
- (B) 1
- (C) 2
- (D) 3
- (E) 4

34) What will be printed when we execute the following code segment?

```
int[] x = {21, 22, 23};  
int[] y = new int[3];  
int[] z = {21, 22, 23};  
  
divider(x, y, z);  
System.out.println(z[2]);
```

- (A) 21
- (B) 22
- (C) 23
- (D) 24
- (E) 25

39) Which of the following statements legally declares, constructs, and initializes an array?

- (A) `int[] myList = {"1", "2", "3"};`
- (B) `int[] myList = {5, 8, 2};`
- (C) `int[] myList = {5, 8, "2"};`
- (D) `int[] myList = {"5", 8, 2};`
- (E) `int[] myList = {5, 8, 2}`

31. A matrix (two-dimensional array) is declared as

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

Consider the following method:

```
public static void changeMatrix(int[] [] mat)
{
    for (int r = 0; r < mat.length; r++)
        for (int c = 0; c < mat[r].length; c++)
            if (r == c)
                mat[r][c] = Math.abs(mat[r][c]);
}
```

If mat is initialized to be

```
-1 -2 -6
-2 -4 5
```

which matrix will be the result of a call to `changeMatrix(mat)`?

(A) 1 -2 -6
-2 4 5

(B) -1 2 -6
2 -4 5

(C) -1 -2 -6
-2 -4 -5

(D) 1 2 -6
2 4 5

(E) 1 2 6
2 4 5

36. Consider a program that has a two-dimensional array `mat` of `int` values. The program has several methods that change `mat` by reflecting elements of `mat` across a mirror placed symmetrically on the matrix. Here are five such methods:

<code>mirrorVerticalLeftToRight</code>	transforms	$\begin{array}{ccc} 2 & 4 & 6 \\ 1 & 3 & 5 \\ 8 & 9 & 0 \end{array}$	to	$\begin{array}{ccc} 2 & 4 & 2 \\ 1 & 3 & 1 \\ 8 & 9 & 8 \end{array}$
<code>mirrorVerticalRightToLeft</code>	transforms	$\begin{array}{ccc} 2 & 4 & 6 \\ 1 & 3 & 5 \\ 8 & 9 & 0 \end{array}$	to	$\begin{array}{ccc} 6 & 4 & 6 \\ 5 & 3 & 5 \\ 0 & 9 & 0 \end{array}$
<code>mirrorHorizontalTopToBottom</code>	transforms	$\begin{array}{ccc} 2 & 4 & 6 \\ 1 & 3 & 5 \\ 8 & 9 & 0 \end{array}$	to	$\begin{array}{ccc} 2 & 4 & 6 \\ 1 & 3 & 5 \\ 2 & 4 & 6 \end{array}$
<code>mirrorHorizontalBottomToTop</code>	transforms	$\begin{array}{ccc} 2 & 4 & 6 \\ 1 & 3 & 5 \\ 8 & 9 & 0 \end{array}$	to	$\begin{array}{ccc} 8 & 9 & 0 \\ 1 & 3 & 5 \\ 8 & 9 & 0 \end{array}$
<code>mirrorDiagonalRightToLeft</code>	transforms	$\begin{array}{ccc} 2 & 4 & 6 \\ 1 & 3 & 5 \\ 8 & 9 & 0 \end{array}$	to	$\begin{array}{ccc} 2 & 4 & 6 \\ 4 & 3 & 5 \\ 6 & 5 & 0 \end{array}$

Consider the following method that transforms the matrix in one of the ways shown above:

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

Which method described above corresponds to `someMethod`?

- (A) `mirrorVerticalLeftToRight`
 (B) `mirrorVerticalRightToLeft`
 (C) `mirrorHorizontalTopToBottom`
 (D) `mirrorHorizontalBottomToTop`
 (E) `mirrorDiagonalRightToLeft`

31) Consider the following code segment.

```
int arr[][] = new int[3][];  
arr[0] = new int[1];  
arr[1] = new int[2];  
arr[2] = new int[3];  
int sum = 0;  
  
for (int i = 0; i < 3; ++i)  
{  
    for (int j = 0; j < i + 1; ++j)  
    {  
        arr[i][j] = j + 1;  
    }  
}  
  
for (int i = 0; i < 3; ++i)  
{  
    for (int j = 0; j < i + 1; ++j)  
    {  
        sum += arr[i][j];  
    }  
}  
System.out.print(sum);
```

What is the output of above code?

- (A) 11
- (B) 10
- (C) 13
- (D) 14
- (E) 0

tes in num a ran-

6. Consider the following instance variables and method `assignValues` in the same class:

```
private int numRows;
private int numCols;
private int [] [] mat;

/** arr has numCols elements */
private void assignValues(int [] arr, int value)
{
    for (int k = 0; k < arr.length; k++)
        arr[k] = value;
}
```

Which of the following code segments will correctly assign `mat` to have the value 100 in each slot? You may assume that the instance variables have all been correctly initialized.

- I for (int row = 0; row < numRows; row++)
 assignValues(mat[row], 100);
- II for (int col = 0; col < numCols; col++)
 assignValues(mat[col], 100);
- III for (int [] row: mat)
 for (int num: row)
 num = 100;

- (A) I only
- (B) II only
- (C) III only
- (D) I and II only
- (E) I and III only

will pass to the

: NEXT PAGE.

GO ON TO THE NEXT PAGE.

27. A square matrix is declared as

```
int[] [] mat = new int[SIZE][SIZE];
```

where SIZE is an appropriate integer constant. Consider the following method:

```
public static void mystery(int[] [] mat, int value, int top, int left,
    int bottom, int right)
{
    for (int i = left; i <= right; i++)
    {
        mat[top][i] = value;
        mat[bottom][i] = value;
    }
    for (int i = top + 1; i <= bottom - 1; i++)
    {
        mat[i][left] = value;
        mat[i][right] = value;
    }
}
```

Assuming that there are no out-of-range errors, which best describes what method mystery does?

- (A) Places value in corners of the rectangle with corners (top, left) and (bottom, right).
- (B) Places value in the diagonals of the square with corners (top, left) and (bottom, right).
- (C) Places value in each element of the rectangle with corners (top, left) and (bottom, right).
- (D) Places value in each element of the border of the rectangle with corners (top, left) and (bottom, right).
- (E) Places value in the topmost and bottommost rows of the rectangle with corners (top, left) and (bottom, right).

28. Which of the following statements about a class SomeClass that implements an interface is (are) true?

- I It is illegal to create an instance of SomeClass.
 - II Any superclass of SomeClass must also implement that interface.
 - III SomeClass must implement every method of the interface.
- (A) None
 - (B) I only
 - (C) II only
 - (D) III only
 - (E) II and III only

15. Consider the following method that will access a square matrix mat:

```

/** Precondition: mat is initialized and is a square matrix.
 */
public static void printSomething(int[] [] mat)
{
    for (int r = 0; r < mat.length; r++)
    {
        for (int c=0; c<=r; c++)
            System.out.print(mat[r][c] + " ");
        System.out.println();
    }
}

```

Suppose mat is originally

```

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

```

After the method call `printSomething(mat)` the output will be

(A) 0 1 2 3
4 5 6 7
3 2 1 0
7 6 5 4

(B) 0
4 5
3 2 1
7 6 5 4

(C) 0 1 2 3
4 5 6
3 2
7

(D) 0
4
3
7

(E) There will be no output. An `ArrayIndexOutOfBoundsException` will be thrown.

16. Cons
there

Met
elem
repr

Me
arr
in
by
co
th
th

[

T

17.

24. Consider the following instance variable and methods in the same class.

```
private int[] [] matrix;

/** Precondition: array.length > 0.
 * @return the largest integer in array
 */
private int max(int[] array)
{ /* implementation not shown */ }

/** @return num1 if num1 >= num2; otherwise return num2
 */
public int max(int num1, int num2)
{ /* implementation not shown */ }
```

Suppose matrix has a current value of

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

What will be returned by the following method call in the same class?

```
max(max(matrix[2]), max(matrix[3]))
```

- (A) 9
- (B) 8
- (C) 7
- (D) 4
- (E) Compile-time error. No value returned.

39. Consider method1 and method2 below, which are identical except for the second last line of code. Each method returns a new matrix based on the input matrix mat.

```
public static int[] [] method1(int[] [] mat)
{
    int numRows = mat.length;
    int numCols = mat[0].length;
    int[] [] newMat = new int[numRows][numCols];
    for (int row = 0; row < numRows; row++)
        for (int col = 0; col < numCols; col++)
            newMat[numRows - row - 1][col] = mat[row][col];
    return newMat;
}

public static int[] [] method2(int[] [] mat)
{
    int numRows = mat.length;
    int numCols = mat[0].length;
    int[] [] newMat = new int[numRows][numCols];
    for (int row = 0; row < numRows; row++)
        for (int col = 0; col < numCols; col++)
            newMat[row][col] = mat[numRows - row - 1][col];
    return newMat;
}
```

Suppose the same input matrix is used for method1 and method2, and the output for method1 is matrix1 while the output for method2 is matrix2. Which is a true statement about matrix1 and matrix2?

- (A) matrix1 is identical to matrix2.
- (B) The rows of matrix1 are the columns of matrix2.
- (C) matrix1 is a reflection of matrix2 across a vertical line on the edge of either matrix.
- (D) matrix1 is a reflection of matrix2 across a horizontal line on the bottom or top edge of either matrix.
- (E) The rows of matrix1 are the rows of matrix2 in reverse order.

MULTIPLE-CHOICE QUESTIONS ON ARRAYS AND ARRAY LISTS

1. Which of the following correctly initializes an array `arr` to contain four elements each with value 0?

I `int[] arr = {0, 0, 0, 0};`
II `int[] arr = new int[4];`
III `int[] arr = new int[4];`
 `for (int i = 0; i < arr.length; i++)`
 `arr[i] = 0;`

- (A) I only
(B) III only
(C) I and III only
(D) II and III only
(E) I, II, and III
2. The following program segment is intended to find the index of the first negative integer in `arr[0] ... arr[N-1]`, where `arr` is an array of `N` integers.

```
int i = 0;
while (arr[i] >= 0)
{
    i++;
}
location = i;
```

This segment will work as intended

- (A) always.
(B) never.
(C) whenever `arr` contains at least one negative integer.
(D) whenever `arr` contains at least one nonnegative integer.
(E) whenever `arr` contains no negative integers.
3. Refer to the following code segment. You may assume that `arr` is an array of `int` values.

```
int sum = arr[0], i = 0;
while (i < arr.length)
{
    i++;
    sum += arr[i];
}
```

Which of the following will be the result of executing the segment?

- (A) Sum of `arr[0], arr[1], ..., arr[arr.length-1]` will be stored in `sum`.
(B) Sum of `arr[1], arr[2], ..., arr[arr.length-1]` will be stored in `sum`.
(C) Sum of `arr[0], arr[1], ..., arr[arr.length]` will be stored in `sum`.
(D) An infinite loop will occur.
(E) A run-time error will occur.

4. Refer to the following code segment. You may assume that array `arr1` contains elements `arr1[0]`, `arr1[1]`, ..., `arr1[N-1]`, where `N = arr1.length`.

```
int count = 0;
for (int i = 0; i < N; i++)
    if (arr1[i] != 0)
    {
        arr1[count] = arr1[i];
        count++;
    }
int[] arr2 = new int[count];
for (int i = 0; i < count; i++)
    arr2[i] = arr1[i];
```

If array `arr1` initially contains the elements 0, 6, 0, 4, 0, 0, 2 in this order, what will `arr2` contain after execution of the code segment?

- (A) 6, 4, 2
- (B) 0, 0, 0, 0, 6, 4, 2
- (C) 6, 4, 2, 4, 0, 0, 2
- (D) 0, 6, 0, 4, 0, 0, 2
- (E) 6, 4, 2, 0, 0, 0, 0

5. Consider this program segment:

```
for (int i = 2; i <= k; i++)
    if (arr[i] < someValue)
        System.out.print("SMALL");
```

What is the maximum number of times that `SMALL` can be printed?

- (A) 0
- (B) 1
- (C) $k - 1$
- (D) $k - 2$
- (E) k

6. What will be output from the following code segment, assuming it is in the same class as the `doSomething` method?

```
int[] arr = {1, 2, 3, 4};
doSomething(arr);
System.out.print(arr[1] + " ");
System.out.print(arr[3]);
...
public void doSomething(int[] list)
{
    int[] b = list;
    for (int i = 0; i < b.length; i++)
        b[i] = i;
}
```

- (A) 0 0
 - (B) 2 4
 - (C) 1 3
 - (D) 0 2
 - (E) 0 3
7. Consider writing a program that reads the lines of any text file into a sequential list of lines. Which of the following is a good reason to implement the list with an `ArrayList` of `String` objects rather than an array of `String` objects?
- (A) The `get` and `set` methods of `ArrayList` are more convenient than the `[]` notation for arrays.
 - (B) The `size` method of `ArrayList` provides instant access to the length of the list.
 - (C) An `ArrayList` can contain objects of any type, which leads to greater generality.
 - (D) If any particular text file is unexpectedly long, the `ArrayList` will automatically be resized. The array, by contrast, may go out of bounds.
 - (E) The `String` methods are easier to use with an `ArrayList` than with an array.
8. Consider writing a program that produces statistics for long lists of numerical data. Which of the following is the best reason to implement each list with an array of `int` (or `double`), rather than an `ArrayList` of `Integer` (or `Double`) objects?
- (A) An array of primitive number types is more efficient to manipulate than an `ArrayList` of wrapper objects that contain numbers.
 - (B) Insertion of new elements into a list is easier to code for an array than for an `ArrayList`.
 - (C) Removal of elements from a list is easier to code for an array than for an `ArrayList`.
 - (D) Accessing individual elements in the middle of a list is easier for an array than for an `ArrayList`.
 - (E) Accessing all the elements is more efficient in an array than in an `ArrayList`.

Refer to the following classes for Questions 9-12.

```
public class Address
{
    private String name;
    private String street;
    private String city;
    private String state;
    private String zip;

    //constructors
    ...

    //accessors
    public String getName()
    { return name; }
    public String getStreet()
    { return street; }
    public String getCity()
    { return city; }
    public String getState()
    { return state; }
    public String getZip()
    { return zip; }
}
```

```
public class Student
{
    private int idNum;
    private double gpa;
    private Address address;

    //constructors
    ...

    //accessors
    public Address getAddress()
    { return address; }
    public int getIdNum()
    { return idNum; }
    public double getGpa()
    { return gpa; }
}
```

9. A client method has this declaration, followed by code to initialize the list:

```
Address[] list = new Address[100];
```

Here is a code segment to generate a list of *names only*.

```
for (Address a : list)
    /* line of code */
```

Which is a correct */* line of code */*?

- (A) System.out.println(Address[i].getName());
- (B) System.out.println(list[i].getName());
- (C) System.out.println(a[i].getName());
- (D) System.out.println(a.getName());
- (E) System.out.println(list.getName());

10. The following code segment is to print out a list of addresses:

```
for (Address addr : list)
{
    /* more code */
}
```

Which is a correct replacement for */* more code */*?

- I System.out.println(list[i].getName());
System.out.println(list[i].getStreet());
System.out.print(list[i].getCity() + ", ");
System.out.print(list[i].getState() + " ");
System.out.println(list[i].getZip());
 - II System.out.println(addr.getName());
System.out.println(addr.getStreet());
System.out.print(addr.getCity() + ", ");
System.out.print(addr.getState() + " ");
System.out.println(addr.getZip());
 - III System.out.println(addr);
- (A) I only
 - (B) II only
 - (C) III only
 - (D) I and II only
 - (E) I, II, and III

29. Consider a class that has this private instance variable:

```
private int[] [] mat;
```

The class has the following method, alter.

```
public void alter(int c)
{
    for (int i = 0; i < mat.length; i++)
        for (int j = c + 1; j < mat[0].length; j++)
            mat[i][j-1] = mat[i][j];
}
```

If a 3×4 matrix mat is

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

then alter(1) will change mat to

(A) 1 5 7 7
2 6 8 8
3 7 9 9

(B) 1 5 7
2 6 8
3 7 9

(C) 1 3 5 7
3 5 7 9

(D) 1 3 5 7
3 5 7 9
3 5 7 9

(E) 1 7 7 7
2 8 8 8
3 9 9 9

28. Which of the following initializes an 8×10 matrix with integer values that are perfect squares? (0 is a perfect square.)

I int[] [] mat = new int[8][10];

II int[] [] mat = new int[8][10];

for (int r = 0; r < mat.length; r++)

for (int c = 0; c < mat[r].length; c++)

mat[r][c] = r * r;

III int[] [] mat = new int[8][10];

for (int c = 0; c < mat[r].length; c++)

for (int r = 0; r < mat.length; r++)

mat[r][c] = c * c;

- (A) I only
(B) II only
(C) III only
(D) I and II only
(E) I, II, and III

30. Consider the following method that will alter the matrix mat:

```
/** @param mat the initialized matrix
 * @param row the row number
 */
public static void matStuff(int[] [] mat, int row)
{
    int numCols = mat[0].length;
    for (int col = 0; col < numCols; col++)
        mat[row][col] = row;
}
```

Suppose mat is originally

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

After the method call `matStuff(mat, 2)`, matrix mat will be

- (A)

```
1 4 9 0
2 7 8 6
2 2 2 2
```
- (B)

```
1 4 9 0
2 2 2 2
5 1 4 3
```
- (C)

```
2 2 2 2
2 2 2 2
2 2 2 2
```
- (D)

```
1 4 2 0
2 7 2 6
5 1 2 3
```
- (E)

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


31. Assume that a square matrix `mat` is defined by

```
int [] [] mat = new int [SIZE] [SIZE];  
//SIZE is an integer constant >= 2
```

What does the following code segment do?

```
for (int i = 0; i < SIZE - 1; i++)  
    for (int j = 0; j < SIZE - i - 1; j++)  
        swap(mat, i, j, SIZE - j - 1, SIZE - i - 1);
```

You may assume the existence of this swap method:

```
/** Interchange mat[a][b] and mat[c][d]. */  
public void swap(int [] [] mat, int a, int b, int c, int d)
```

(A) Reflects `mat` through its major diagonal. For example,

```
2 6      2 4  
  →  
4 3      6 3
```

(B) Reflects `mat` through its minor diagonal. For example,

```
2 6      3 6  
  →  
4 3      4 2
```

(C) Reflects `mat` through a horizontal line of symmetry. For example,

```
2 6      4 3  
  →  
4 3      2 6
```

(D) Reflects `mat` through a vertical line of symmetry. For example,

```
2 6      6 2  
  →  
4 3      3 4
```

(E) Leaves `mat` unchanged.