**Objective 1 - Data Structure Definitions**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
</table>
| 01. | A data structure is a data type  
   | (A) with a single value.  
   | (B) with two or more values.  
   | (C) with one or more simple data types.  
   | (D) whose components are smaller data structures and/or simple data types.  
| 02. | Which is the first historical data structure?  
   | (A) object  
   | (B) record  
   | (C) array  
   | (D) file  
| 03. | An array is a  
   | (A) data structure with one, or more, elements of the same type.  
   | (B) data structure with LIFO access.  
   | (C) data structure, which allows transfer between internal and external storage.  
   | (D) data structure with one, or more, elements, called fields, of the same or different data types.  
| 04. | A record is a  
   | (A) data structure with one, or more, elements of the same type.  
   | (B) data structure with LIFO access.  
   | (C) data structure, which allows transfer between internal and external storage.  
   | (D) data structure with one, or more, elements, called fields, of the same or different data types.  

<table>
<thead>
<tr>
<th>Question</th>
<th>Correct Answer</th>
</tr>
</thead>
<tbody>
<tr>
<td>05. A file is a</td>
<td>(C) data structure, which allows transfer between internal and external storage.</td>
</tr>
<tr>
<td>06. A stack is a</td>
<td>(B) data structure with LIFO access.</td>
</tr>
<tr>
<td>07. Data structures are defined by</td>
<td>(C) both the data storage and the data access.</td>
</tr>
<tr>
<td>08. The array was first introduced by the __________ programming language.</td>
<td>(A) FORTRAN</td>
</tr>
</tbody>
</table>
Objective 2 - One-Dimensional Array Declaration and Access

09. Consider the following program segment.

```java
int list[ ];
list = new int[100];
```

How many integers can be stored in the `list` array

(A) 99  
(B) 100  
(C) 101  
(D) 100 initial integers plus any additional integers required during program execution

10. Consider the two program segments below.

Segment1

```java
int list[ ];
list = new int[100];
```

Segment2

```java
int list[ ] = new int[100];
```

Which of the following is a true statement about the comparison of Segment1 and Segment2?

(A) Segment1 declares `list` correctly. Segment2 declares `list` incorrectly.  
(B) Segment1 declares `list` incorrectly. Segment2 declares `list` correctly.  
(C) Both Segment1 and Segment2 declare `list` correctly.  
(D) Both Segment1 and Segment2 declare `list` incorrectly.

11. Consider the program segment below.

```java
double grades[ ];
grades = new double[50];
```

What is the index range capable of accessing an element of the `grades` array?

(A) 0..49  
(B) 1..49  
(C) 0..50  
(D) 1..50
Objective 3 - Accessing Elements In An Array

12. What is the output of program **Java1012.java** below?

```java
public class Java1012 {
    public static void main(String[] args) {
        int list[];
        list = new int[10];
        for (int k = 0; k < 10; k++)
            list[k] = 0;
        for (int k = 0; k < 10; k++)
            System.out.print(list[k] + " ");
        System.out.println();
    }
}
```

###

(A) 0 0 0 0 0 0 0 0 0 0
(B) 0 1 2 3 4 5 6 7 8 9
(C) 1 2 3 4 5 6 7 8 9 10
(D) 0 0 0 0 0 0 0 0 0 0

13. What is the output of program **Java1013.java** below?

```java
public class Java1013 {
    public static void main(String args[ ]) {
        char list[] = new char[5];
        for (int k = 0; k < 5; k++)
            list[k] = 'Q';
        for (int k = 0; k < 5; k++)
            System.out.print(list[k] + " ");
        System.out.println();
    }
}
```

###

(A) Q Q Q Q Q
(B) Q R S T U
(C) Q Q Q Q Q
(D) Q Q Q Q Q
14. What is the output of program `Java1014.java` below?

```java
public class Java1014 {
    public static void main(String args[]) {
        int list[] = {1,2,3,4,5};
        for (int k = 1; k < list.length; k++)
            System.out.println("list[" + k + "] = " + list[k]);
    }
}
```

(A) list[0] = 0
list[1] = 1
list[2] = 2
list[3] = 3
list[4] = 4

(B) list[0] = 1
list[1] = 2
list[2] = 3
list[3] = 4
list[4] = 5

(C) list[1] = 1
list[2] = 2
list[3] = 3
list[4] = 4
list[5] = 5

(D) list[1] = 2
list[2] = 3
list[3] = 4
list[4] = 5

(E) Compile Error
public class Java1015
{
    public static void main(String args[ ])
    {
        int list[ ] = {1,2,3,4,5};
        for (int k = 1; k <= 5; k++)
            System.out.println("list[" + k + "] = " + list[k]);
    }
}

(A)  list[0] = 0
     list[1] = 1
     list[2] = 2
     list[3] = 3
     list[4] = 4

(B)  list[0] = 1
     list[1] = 2
     list[2] = 3
     list[3] = 4
     list[4] = 5

(C)  list[1] = 1
     list[2] = 2
     list[3] = 3
     list[4] = 4
     list[5] = 5

(D)  list[1] = 2
     list[2] = 3
     list[3] = 4
     list[4] = 5

### (E)  Compile Error
16. What is the output of program `Java1016.java` below?

```java
public class Java1016 {
    public static void main(String args[]) {
        int list[] = {1,2,3,4,5};
        for (int k = list.length-1; k >= 0; k--)
            System.out.println("list[" + k + "] = " + list[k]);
    }
}
```

###
(A) `list[4] = 5`  
`list[3] = 4`  
`list[2] = 3`  
`list[1] = 2`  
`list[0] = 1`  

(B) `list[5] = 5`  
`list[4] = 4`  
`list[3] = 3`  
`list[2] = 2`  
`list[1] = 1`  

(C) `list[5] = 4`  
`list[4] = 3`  
`list[3] = 2`  
`list[2] = 1`  
`list[1] = 0`  

(D) `list[4] = 1`  
`list[3] = 2`  
`list[2] = 3`  
`list[1] = 4`  

(E) Compile Error
17. What is the FIRST and LAST output from this program segment?

```java
int IntNum[] = new int[100];
int J;
for (J=0; J<100; J++)
    IntNum[J] = J;
for (J=0; J<100; J++)
    System.out.println(IntNum[J]);
```

(A) 0 and 100
(B) 0 and 99
(C) 1 and 100
(D) 1 and 99
(E) ArrayIndexOutOfBoundsException error

18. What is the FIRST and LAST output from this program segment?

```java
int IntNum[] = new int[100];
int J;
for (J=1; J<=100; J++)
    IntNum[J] = J;
for (J=1; J<=100; J++)
    System.out.println(IntNum[J]);
```

(A) 0 and 100
(B) 0 and 99
(C) 1 and 100
(D) 1 and 99
(E) ArrayIndexOutOfBoundsException error

19. What is the FIRST and LAST output from this program segment?

```java
int IntNum[] = new int[100];
int J;
for (J=0; J<100; J++)
    IntNum[J] = 100-J;
for (J=85; J>=15; J--)
    System.out.println(IntNum[J]);
```

(A) FIRST: 0 LAST: 99
(B) FIRST: 99 LAST: 0
(C) FIRST: 15 LAST: 85
(D) FIRST: 85 LAST: 15
(E) ArrayIndexOutOfBoundsException error

20. What is the output from this program segment?

```java
int IntNum[] = new int[10];
int J;
for (J=0; J<10; J++)
    if (J < 2)
        IntNum[J] = J;
    else
        IntNum[J] = IntNum[J-1] + IntNum[J-2];
for (J=0; J<10; J++)
    System.out.print(IntNum[J] + " ");
```

(A) 1 2 3 4 5 6 7 8 9 10
(B) 0 1 2 3 4 5 6 7 8 9
(C) 1 1 2 3 5 8 13 21 34 55
(D) 0 1 1 2 3 5 8 13 21 34
(E) 0 1 1 2 3 5 8 13 21 34 55
Objective 4 – Initialized Arrays and length

Use this program segment to answer questions 21-24.

```java
boolean George[] = new boolean[15];
int J;

System.out.println(George.length); // Question #21

for (J=0; J<15; J++)
    if (J == 0)
        George [J] = (J==0);
    else
        George [J] = !George[J-1];

System.out.println(George[7]); // Question #22
System.out.println(George[8]); // Question #23
System.out.println(George[15]); // Question #24
```

<table>
<thead>
<tr>
<th>Question</th>
<th>Output</th>
<th>Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>21.</td>
<td></td>
<td>(A) true</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(B) false</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(C) 14</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(D) 15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(E) ArrayIndexOutOfBoundsException error</td>
</tr>
<tr>
<td>22.</td>
<td></td>
<td>(A) true</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(B) false</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(C) 14</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(D) 15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(E) ArrayIndexOutOfBoundsException error</td>
</tr>
<tr>
<td>23.</td>
<td></td>
<td>(A) true</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(B) false</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(C) 14</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(D) 15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(E) ArrayIndexOutOfBoundsException error</td>
</tr>
<tr>
<td>24.</td>
<td></td>
<td>(A) true</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(B) false</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(C) 14</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(D) 15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(E) ArrayIndexOutOfBoundsException error</td>
</tr>
</tbody>
</table>
Objective 5 – Random Array Elements

25. What is the output of the following program segment?

```java
import java.util.Random;

public class Java1025 {
    public static void main(String args[]) {
        int list[] = new int[20];
        Random random = new Random(12345);
        for (int k = 0; k < 20; k++)
            list[k] = random.nextInt(900) + 100;
        for (int k = 0; k < 20; k++)
            System.out.println(list[k]);
        System.out.println();
    }
}
```

(A) A set of integers in the [900..12345] range
(B) A set of integers in the [100..12345] range
(C) A set of integers in the [0..899] range
(D) A set of integers in the [100..999] range
(E) A set of integers in the [100..1000] range
26. What can be determined about the contents of the list array?

import java.util.Random;

public class Java1026 {
    public static void main(String args[]) {
        int list[] = {0,1,2,3,4,5,6,7,8,9};
        Random r = new Random();
        for (int k = 0; k < list.length; k++)
            list[r.nextInt(10)] = k;
        System.out.println();
    }
}

(A) The original list array contains {0,1,2,3,4,5,6,7,8,9} and then every element of the list array is changed to a random value in the [0..9] range.

(B) The original list array contains {0,1,2,3,4,5,6,7,8,9} and then random elements of the list array are changed to the current value of k.

(C) The original list array contains {0,1,2,3,4,5,6,7,8,9} and stays unchanged throughout the program execution.

(D) The original list array contains {0,1,2,3,4,5,6,7,8,9} and then random elements of the list array are changed to random values.

(E) Without knowing the seed value of the Random object, nothing can be stated about the values of any element in the list array.
27. What can be determined about the contents of the list array?

```java
import java.util.Random;

public class Java1027
{
    public static void main(String args[])
    {
        int list[] = {0,1,2,3,4,5,6,7,8,9};
        Random r = new Random();
        for (int k = 0; k < list.length; k++)
            list[k] = r.nextInt(10);
        System.out.println();
    }
}
```

### (A) The original list array contains {0,1,2,3,4,5,6,7,8,9} and then every element of the list array is changed to a random value in the [0..9] range.

(B) The original list array contains {0,1,2,3,4,5,6,7,8,9} and then random elements of the list array are changed to the current value of k.

(C) The original list array contains {0,1,2,3,4,5,6,7,8,9} and stays unchanged throughout the program execution.

(D) The original list array contains {0,1,2,3,4,5,6,7,8,9} and then random elements of the list array are changed to random values.

(E) Without knowing the seed value of the Random object, nothing can be stated about the values of any element in the list array.
What can be determined about the contents of the list array?

```java
import java.util.Random;

public class Java1028 {
    public static void main(String args[]) {
        int list[] = {0,1,2,3,4,5,6,7,8,9};
        Random r = new Random();
        for (int k = 0; k <= 5; k++)
            list[k] = r.nextInt(10);
        System.out.println();
    }
}
```

(A) The original list array contains {0,1,2,3,4,5,6,7,8,9} and then every element of the list array is changed randomly to a value in the \([0..9]\) range.

(B) The original list array contains {0,1,2,3,4,5,6,7,8,9} and then random elements of the list array are changed to the current value of \(k\).

(C) The original list array contains {0,1,2,3,4,5,6,7,8,9} and stays unchanged throughout the program execution.

(D) The original list array contains {0,1,2,3,4,5,6,7,8,9} and then the first five elements of the list array are changed to random values.

(E) The last four elements of the list array remain unchanged with values 6,7,8,9.
29. What is true about the new for loop?

(A) It is called the "for..each" loop.
(B) It requires that Java 5.0 is installed.
(C) It is not possible to access specific array elements.
### (D) All of the above

30. Which of the following statement displays the list elements correctly?

```java
int list[] = {11,22,33,44,55,66,77,88,99};

(A) for (int k=0; list item; k++)
   System.out.print(item + " ");
### (B) for (int item: list)
   System.out.print(item + " ");

(C) for (int k=0; int item; k++)
   System.out.print(item + " ");

(D) for (int k=0; list item; k++)
   System.out.print(item[k] + " ");
```

31. Rewrite the old for loop program segment below with the new for loop.

```java
int list[] = {1,2,3,4,5,6};
for (int k = 0; k < list.length; k++)
   System.out.println(list[k]);

### (A) for (int number: list)
   System.out.print(number + " ");

(B) for (int number: list.length)
   System.out.print(number + " ");

(C) for (int k = 0; number: list)
   System.out.print(number[k]);

(D) This program segment cannot be converted to the new for loop.
```
32. Rewrite the old for loop program segment below with the new for loop.

   for (int k = 0; k < 10; k++)
      System.out.println(k);

   (A) for (int number: k)
       System.out.print(number + " ");

   (B) for (int number: k.length)
       System.out.print(k + " ");

   (C) for (int k = 0; number: list)
       System.out.print(number[k]);

   ### (D) This program segment cannot be converted to the new for loop.

---

**Objective 7 - Java Static Two-Dimensional Arrays**

33. Which of the following statements correctly declares a two-dimensional integer array?

   (A) int Matrix[ ] = new int[5,4];

   (B) int Matrix[ ];
      Matrix = new int[5,4];

   (C) int Matrix[ ][ ] = new int[5][4];

   (D) int Matrix[ ][ ];
      Matrix = new int[5][4];

   ### (E) Both choices C and D
34. What is the output of the program below?

```java
public class Java1034 {
    public static void main(String[] args) {
        int matrix[][];
        matrix = new int[3][4];
        for (int p = 0; p < 3; p++) {
            for (int q = 0; q < 4; q++)
                System.out.print(matrix[p][q] + " ");
            System.out.println();
        }
        System.out.println();
    }
}
```

(A) 0 0 0 0 0 0 0 0 0 0 0 0
(B) 0 0 0 0 0
(C) 0
(D) Compile Error

35. Consider the `mambo` object declaration below.

```java
double mambo[][ ];
mambo = new double[4][5];
int r; // row index on mambo
int c; // column index of mambo
```

Which of the following statements stores the column length of `mambo`?

(A) `mambo.length`
(B) `mambo.getRowLength`
(C) `mambo[r].length`
(D) `mambo[c].length`
36. What is the output of the program below?

```java
public class Java1036 {
    public static void main(String args[]) {
        int matrix[][];
        matrix = new int[3][4];
        for (int p = 0; p < 3; p++) {
            for (int q = 0; q < 4; q++)
                System.out.print(matrix[q][p] + " ");
            System.out.println();
        }
    }
}
```

(A) 0 0 0 0  
(B) 0 0 0 0  
(C) 0 0 0 0  
(D) Error message

37. Consider the `mambo` object declaration below.

```java
double mambo[][];
mambo = new double[4][5];
int r; // row index on mambo
int c; // column index of mambo
```

Which of the following statements stores the `row length` of `mambo`?

(A) `mambo.length`  
(B) `mambo.rowLength`  
(C) `mambo[r].length`  
(D) `mambo[c].length`
Use the program below for questions 38-40. Each question will provide a different implementation of the `createSquare` method. The output shown is formatted for ease of reading. Technically, the columns will not line up so nicely.

```java
public class Java3840 {
    public static void main (String args[]) {
        int square[][] = new int[5][5];
        createSquare(square);
        displaySquare(square);
    }

    public static void createSquare(int[][] square) {
    }

    public static void displaySquare(int[][] square) {
        for (int r = 0; r < 5; r++) {
            for (int c = 0; c < 5; c++)
                System.out.print(square[r][c] + " ");
            System.out.println();
        }
    }
}
```
38. What will be the output of program Java3840.java with the createSquare implementation below?

```java
public static void createSquare(int[][] square)
{
    int size = square.length;
    for (int r = 0; r < size; r++)
    {
        int q = 1;
        for (int c = r; c < size; c++)
        {
            square[r][c] = q;
            q++;  
        }
    }
}
```

(A) 0 1 2 3 4
    1 2 3 4 5
    2 3 4 5 6
    3 4 5 6 7
    4 5 6 7 8

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

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

(D) 1 2 3 4 5
    6 7 8 9 10
    11 12 13 14 15
    16 17 18 19 20
    21 22 23 24 25
39. What will be the output of program `Java3840.java` with the `createSquare` implementation below?

```java
public static void createSquare(int[][] square)
{
    int size = square.length;
    for (int r = 0; r < size; r++)
    {
        int q = r;
        for (int c = 0; c < size; c++)
        {
            square[r][c] = q + c;
        }
    }
}

###
(A) 0 1 2 3 4
    1 2 3 4 5
    2 3 4 5 6
    3 4 5 6 7
    4 5 6 7 8

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

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

(D) 1 2 3 4 5
    6 7 8 9 10
    11 12 13 14 15
    16 17 18 19 20
    21 22 23 24 25
```
40. What will be the output of program Java3840.java with the createSquare implementation below?

```java
public static void createSquare(int[][] square)
{
    int size = square.length;
    int r = 0;
    int c = size / 2;
    square[r][c] = 1;
    for (int k = 2; k <= size*size; k++)
    {
        if (k % size == 1)
            r++;
        else
        {
            r--;  c++;
        }
        if (r < 0)  r = size-1;
        if (c == size)  c = 0;
        square[r][c] = k;
    }
}
```

(A) 1  6 11 16 21  
     2  7 12 17 22  
     3  8 13 18 23  
     4  9 14 19 24  
     5 10 15 20 25  

(B) 11 18 25   2 9  
     10 12 19 21 3  
     4  6 13 20 22  
     23 5  7 14 16  
   17 24  1  8 15  

### (C) 17 24  1  8 15  
       23 5  7 14 16  
       4  6 13 20 22  
      10 12 19 21 3  
     11 18 25   2 9  

(D) 1  2  3  4  5  
     6  7  8  9 10  
    11 12 13 14 15  
    16 17 18 19 20  
   21 22 23 24 25  

```
Use the class and program segment below for questions 41-42.

```java
class Student {
    private String name;
    private int age;

    public Student(String n, int a) {
        name = n;
        age = a;
    }

    public void showData() {
        System.out.println("Name: "+ name);
        System.out.println("Age: " + age);
        System.out.println();
    }
}

Student students[] = new Student[numStudents];
for (int index = 0; index < numStudents; index++) {
    System.out.print("Enter student's name ==> ");
    String name = stringInput.nextLine();
    System.out.print("Enter student's age ==> ");
    int age = intInput.nextInt();
    students[index] = new Student(name, age);
}
```

41. The relationship between the Students object and the Student objects is an example of

(A) inheritance.
(B) an initializer list.
(C) composition.
(D) nested arrays.
42. Which of the following describes the relationship of the objects?

### (A) The program uses a **Students** array object of **Student** objects.
(B) The program uses a **Student** array object of **Students** objects.
(C) The program uses a **Student** object of **Student** array objects.
(D) The program uses a **Students** object of **Students** array objects.

Use the class and program segment below for questions 43-44.

class MyList
{
    private int intArray[];
    private int size;

    public MyList(int s)
    {
        intArray = new int[size];
    }
}

MyList list = new MyList(10);

43. Describes the result of executing the statement **MyList list = new MyList(10);**

(A) It instantiates a **list** object.
(B) It instantiates an **IntArray** object.
(C) It allocates spaces for ten **int** values.
### (D) All of the above.

44. Which of the following describes the relationship of the objects?

(A) The program uses a **MyList** array object of **int** values.
(B) The program uses a **MyList** object of **IntArray** values.
### (C) The program uses a **MyList** object with an **IntArray** of **int** values.
(D) The program uses a **IntArray** object of **MyList** values.
### Objective 9 - Two-Dimensional Arrays and length

(This topic was also included in the earlier two-dimensional array section; this is a strict focus on length)

<table>
<thead>
<tr>
<th>Question</th>
<th>Statement</th>
</tr>
</thead>
</table>
| 45. | Consider the following two-dimensional array declaration.  
\[
\text{int}[][] \text{ matrix} = \text{new int}[4][4];
\]  
Which of the following statements will assign the correct size to **rowSize**?  

(A) \[ \text{int rowSize} = \text{matrix}.\text{length}; \]  
(B) \[ \text{int rowSize} = \text{matrix}[0].\text{length}; \]  
(C) \[ \text{int rowSize} = \text{matrix}[1].\text{length}; \]  
(D) \[ \text{int rowSize} = \text{matrix}[2].\text{length}; \]  
### (E) All of the above

<table>
<thead>
<tr>
<th>Question</th>
<th>Statement</th>
</tr>
</thead>
</table>
| 46. | Consider the following two-dimensional array declaration.  
\[
\text{int}[][] \text{ matrix} = \text{new int}[4][4];
\]  
Which of the following statements will assign the correct size to **colSize**?  

(A) \[ \text{int colSize} = \text{matrix}.\text{length}; \]  
(B) \[ \text{int colSize} = \text{matrix}[0].\text{length}; \]  
(C) \[ \text{int colSize} = \text{matrix}[1].\text{length}; \]  
(D) \[ \text{int colSize} = \text{matrix}[2].\text{length}; \]  
### (E) All of the above
47. Consider the following two-dimensional array declaration.

```java
int[][] matrix = new int[4][5];
```

Which of the following statements will assign the correct size to `colSize`?

(A) `int colSize = matrix[0].length;`

(B) `int colSize = matrix[1].length;`

(C) `int colSize = matrix[2].length;`

(D) `int colSize = matrix[3].length;`

(E) All of the above

48. What will be printed by the following program statement?

```java
System.out.println(matrix[0].length);
```

(A) The number of rows in a two-dimensional "square" static array.

(B) The number of columns in a two-dimensional "non-ragged" array.

(C) The number of columns in the top row of a two-dimensional static array.

(D) The number of columns in the row with index[0] of a two-dimensional array.

(E) All of the above
49. Consider the following program.

```java
public class Question49 {
    public static void main (String args[]) {
        int p = 10;
        int q = 20;
        swap(p,q);
        System.out.println(p + " " + q);
    }

    public static void swap(int x, int y) {
        int temp = x;
        x = y;
        y = temp;
    }
}
```

What is printed as a result of executing the program?

###
(A) 10 20
(B) 20 10
(C) 10 10
(D) 20 20
(E) 0 0
50. Consider the following program.

```java
public class Question50 {

    public static void main (String args[]) {
        int[] list = {1,2,3,4,5,6,7,8,9};
        swap(list,3,4);
        System.out.println(list[3] + " " + list[4]);
    }

    public static void swap(int[] x, int p, int q) {
        int temp = x[p];
        x[p] = x[q];
        x[q] = temp;
    }

}
```

What is printed as a result of executing the program?

(A) 3 4  
(B) 4 3  
(C) 4 5  
(D) 5 4  
(E) ArrayIndexOutOfBoundsException