|  |  |
| --- | --- |
| Question 01  What is output by the code at right?  A. 4 B. 11  C. 9 D. 5  E. 10 | int[]gRay = new int[10];  gRay[0] = 33;  gRay[1] = 14;  gRay[2] = 37;  gRay[3] = 11;  gRay[4] = 27;  out.println(gRay.length); |
| Use the following code for questions 02-04  int[] array = {7,8,10,11,4,3};  array[array[0]/2]=15;  array[array[4]+1]=9;  array[array.length/2-1]=5;  array[1]=array[0]+4; | |
| Question 02  After running the code shown above, what is output by the following code?  System.out.println(array[2]);  A. 5 B. 11 C. 7 D. 9 E. 15 | |
| Question 03  After running the code shown above, what is output by the following code?  System.out.println (array[0]);  A. 5 B. 11 C. 7 D. 9 E. 15 | |
| Question 04  After running the code shown above, what is output by the following code?  System.out.println (array[5]);  A. 5 B. 11 C. 7 D. 9 E. 15 | |

|  |  |
| --- | --- |
| Question 05  Which of the following ArrayList instantiations would create an ArrayList that could store doubles?  A. ArrayList<double> aList = new ArrayList<double>();  B. ArrayList<Integer> aList = new ArrayList<Integer>();  C. ArrayList<Double> aList = new ArrayList<Double>();  D. ArrayList<Boolean> aList = new ArrayList<Boolean>();  E. None of these | |
| Question 06  What is the output?  A. [one, two, three]  B. [one]  C. [three, two, one]  D. [three]  E. None of these | ArrayList<String> hList;  hList = new ArrayList<String>();  hList.add(0,"one");  hList.add(0,"two");  hList.add(0,"three");  System.out.println(hList); |
| Question 07  What is the output?  A. [2.5, 5.3, 9.0]  B. [9.0, 5.3, 2.5]  C. [9.0]  D. [2.5]  E. None of these | ArrayList<Double> mList;  mList = new ArrayList<Double>();  mList.add(2.5);  mList.add(5.3);  mList.add(9.0);  System.out.println(mList); |
| Question 08  What is the output?  A. one two three  B. three two one  C. one D. error E. None of these | ArrayList objList = new ArrayList();  objList.add("one");  objList.add("two");  objList.add("three");  for(int i=objList.size(); i>=0; i--)  System.out.print(objList.get(i) + " ");  System.out.println(); |
| Question 09  What is the output?  A. 2 B. 1  C. 0 D. 4  E. 3 | ArrayList funList = new ArrayList();  funList.add(0,"one");  funList.add("two");  funList.add(0,"three");  funList.add("four");  funList.add(0,"five");  System.out.println(funList.indexOf("four")); |

|  |  |
| --- | --- |
| Question 10  **Write the code below** to correctly fill  blank **<\*1>**  to assign the first letter of each String to variable letter? | List<String> ws = new ArrayList<String>();  // code not seen to populate ws with Strings  int count=0;  for(int x=0; x<ws.size(); x++)  {  char letter = **<\*1>**  if(letter == 's')  count++;  } |
| Question 11  Which of the following would fill blank **<\*1>** to add a new It object to itListOne?  A. itListOne.add(56);  B. itListOne.add(It(56));  C. itListOne.add(new It(56));  D. itListOne.add(new 56);  E. more than one of these | public class It{  private int stuff;  public It(int x){  stuff=x;  }  public String toString(){  return ""+stuff;  }  }  **//code in client class**  ArrayList<It> itListOne;  itListOne = new ArrayList<It>();  **<\*1>** |
| Question 12  What is the output?  A. [one]  B. [two, one, one] C. [two, one]  D. [one, two, one]  E. [two] | ArrayList<String> whatList;  whatList = new ArrayList<String>();  whatList.add("one");  whatList.add("two");  whatList.add("one");  whatList.add("one");  whatList.add("one");  int spot=0;  while(spot<whatList.size())  {  if(whatList.get(spot).equals("one"))  whatList.remove(spot);  else  spot++;  }  out.println(whatList); |

|  |
| --- |
| Question 13  Consider the following instance variable and incomplete method.  The method sum should sum up all of the values in list.  private ArrayList<Double> list; //assume the array contains values  public double sum()  {  double sum = 0;  /\* code \*/  return sum;  }  Which of the following code segments shown below could be used to replace /\* code \*/ so that sum will work as intended?  I. for ( int i = 0; i < list.size(); i++)  sum = sum + list.get(i);  II. for ( int i = list.size()-1; i > -1; i--)  sum = sum + list.get(i);  III. for ( int i = 0; i < list.size(); i++)  sum = sum + i;  A. I only  B. II only  C. III only  D. I and III only  E. I and II only |

|  |
| --- |
| Question 14  Consider the following instance variable and incomplete method.  The method sum should sum up every other value in the list.  private List<Double> list; //assume the array contains values  public double sumEveryOther()  {  double sum = 0;  /\* code \*/  return sum;  }  Which of the following code segments shown below could be used to replace /\* code \*/ so  that sumEveryOther will sum up every other value as intended?  I. for ( int i = 0; i < list.size()/2; i++)  sum = sum + list.get(i);  II. for ( int i = list.size()-1; i > -1; i = i - 2)  sum = sum + list.get(i);  III. for ( int i = 0; i < list.size(); i = i + 2)  sum = sum + list.get(i);  A. I only  B. II and III only  C. III only  D. I and III only  E. I, II, and III |

15. Consider the following code segment.

int[][] M;

M = initialize(); // initialize() will make M a square matrix and initialize its elements

int sum = 0;

int k;

for (k = 0; k < M.length - 1 ; k++)

{

sum += M[k+1][M.length – k – 1];

}

Assume that after the call to Initialize, M represents the matrix shown below

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **0** | **1** | **2** | **3** |
| **0** | 1 | 1 | 1 | 1 |
| **1** | 1 | 2 | 3 | 4 |
| **2** | 2 | 2 | 2 | 2 |
| **3** | 2 | 4 | 6 | 8 |

What value will sum contain after the code segment is executed?

(a) 4

(b) 8

(c) 10

(d) 20

(e) 42

16. Consider the following code segment.

int row, col;

int sum = 0;

int[][] A;

A = initialize(); // sizes A and initializes its elements

for (row = 0; row < A.length; row++)

{

for (col = 0; col < A[0].length; col++)

{

sum += A[row][col];

}

}

Which of the following best describes the result of executing the code segment?

(a) Each element in the two-dimensional array A contains the value 0.

(b) Each element in the two-dimensional array A contains the sum of its row number and its column number

(c) Each element in the two-dimensional array A contains the sum of all preceding elements in two-dimensional array A.

(d) The variable sum contains the sum of the values in the two-dimensional array A.

(e) The variable sum contains the value row \* col.

17. Consider the following code.

int[][] M;

M = initialize(); // sizes M to be a square matrix and initializes its elements

Which of the following code segments correctly sets a diagonal of the two-dimensional array M to contain all zeroes?

**WRITE THE CODE TO FIND THE MAJOR DIAGONAL (0,0), (1,1), … (N,N)**

(d) int row;

for (row = 0; row < M.length; row++)  
 {

M[row][row] = 0;

}

}

18. Consider the following method:

public boolean somethingDifferent(boolean p, boolean q)

{

return ((p || q) && !(p && q));

}

What does method somethingDifferent return?

a) somethingDifferent always returns false

b) somethingDifferent always returns true

c) somethingDifferent returns true whenever p is false

d) somethingDifferent returns true whenever q is false

e) somethingDifferent returns true whenever p is not equal to q

}

19. Consider designing a data structure to represent the positions of 50 game pieces on a 100 x 100 game board. (The position of a game piece is the row and column number of the square that it is on.) Two alternatives are described below.

Method 1: Use a two-dimensional array of boolean values indexed by row and column number, where each array element represents one square of the game board. If there is a game piece on that square, then the array element is true; otherwise, the array element is false.

Method 2: Use a one-dimensional array in which each element represents the position of one game piece (i.e., the row and column number of the square that it is on).

Which of the following is **true**?

(a) Method 1 is not suitable if two game pieces can occupy the same square of the game board.

(b) Method 2 is not suitable if two game pieces can occupy the same square of the game board.

(c) Printing the positions of all game pieces can be done more efficiently by using Method 1 than by using Method 2.

(d) Determining whether there is a game piece on a particular square (given the row and column numbers) can be done more efficiently by using Method 2 than by using Method 1.

(e) Removing the game piece from a particular square (given the row and column numbers) can be done more efficiently by using Method 2 than by using Method 1.

1. What is the output of the following code segment?

String str1 = "A";

String str2 = str1;

str2 += "B";

System.out.println(str1 + str2);

A. AB

B. BA

C. AAB

D. ABAB

**DIRECTIONS :** Fill in each blank with the correct answer/output. Assume each statement happens in order and that one statement may affect the next statement.

public class Grade{

//private double grade;

public Grade(double g){

grade = g;

}

public char getLetter(){

//code not shown

//converts grade to a a letter grade – e.g. 93🡺A

}

public String toString(){

return ""+String.format("%.2f",grade);

}

}

//test code in a client class

//instantiate an ArrayList of Grade references

//write the code to load in 8 Grade objects, randomly chosen between 60 and 100, inclusive – use a for loop

//write the code to print out each of the 8 Grades

//write the code to print out each of the 8 Grades as a letter

**PART 1 : Show the output of each block of code below.**

1. What is the output?

List<Integer> list = new ArrayList<Integer>();

list.add(3);

list.add(6);

list.add(5);

list.add(8);

list.add(12);

int count=0;

for(int spot=0; spot<list.size(); spot++){

if(list.get(spot)%2==0)

list.remove(spot);

}

out.println(list);

**PART 2 :** Fill in the method below with the appropriate code.

5. Take a 2D Array and switch row and column reverse order and return the new 2D Array

2 5

3 1

4 6

Becomes

2 3 4

5 1 6

public static int[][] switchAround (int[][] matrix)

6. Create an ArrayList of the first N numbers in the Fibonacci sequence (1, 1, 2, 3, 5, 8, …) and return the ArrayList

public static List<Integer> fibo(int n)