AP Computer Science \_\_\_\_\_\_\_ Name \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tests: 2D array 35 points Date \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Period \_\_\_\_\_\_

ArrayList, Tricky Polymorphism

AP Quick Reference Sheet Allowed

**Multiple Choice (1 point each): Choose the best answer and circle it on your quiz.**

1. Consider the following code segment.

ArrayList<String> letters = new ArrayList<String>();

letters.add(“H”);

letters.add(“J”);

letters.add(“K”);

letters.add(1, “R”);

letters.remove(2);

letters.set(0, letters.get(2));

What is contained in the letters ArrayList after the segment is run?

1. H J K R
2. K R K
3. R J R
4. H J R
5. K R J

2.

Assume that the variable myList is an ArrayList that contains at least two objects. Which of the following code segments moves the first object to the end of the list?

1. myList.add(0, myList.remove(0));
2. myList.add(myList.remove(myList.size() – 1));
3. myList.add(myList.get(0));
4. myList.add(myList.remove(0));
5. myList.add(myList.size(), myList.get(0));

3.

Assume that the variable numbers is an ArrayList of Integers.

Consider the following code segment.

boolean flag = true;

for (int val : numbers) {

flag = flag && (val > 0);

}

What does this code segment do?

1. Sets flag to true if the last value is positive.
2. Always sets flag to false.
3. Sets flag to true if any value in numbers is positive.
4. Always sets flag to true.
5. Sets flag to true if every value in numbers is positive.

4. Consider the following code segment.

int[][] values = {{1, 2, 3}, {4, 5, 6}};

int x = 0;

for (int j = 0; j < values.length; j++) {

for (int k = 0; k < values[0].length; k++) {

if (k == 0) {

values[j][k] \*= 2;

}

x += values[j][k];

}

}

What is the value of x after the code is executed?

A. 7

B. 17

C. 21

D. 26

E. 27

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

private int[][] mat;

The class has the following method, modify.

public void modify(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 x 4 matrix mat is

1 3 5 7

2 4 6 8

3 5 7 9

Then modify(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

**6 – 10 (1 point each) Assuming that the following classes have been defined:**

public class A

{

public void method1()

{

System.out.print("A1 ");

}

public void method3()

{

System.out.print("A3 ");

}

}

public class B extends A

{

public void method3()

{

System.out.print("B3 ");

super.method3();

}

}

public class C extends B

{

public void method2()

{

System.out.print("C2 ");

method1();

}

}

public class D extends B

{

public void method1()

{

System.out.print("D1 ");

super.method1();

}

public void method3()

{

System.out.print("D3 ");

super.method3();

}

}

And assuming the following variables have been defined:

A var1 = new C();

B var2 = new B();

A var3 = new D();

D var4 = new D();

Object var5 = new A();

In the table below, indicate in the right-hand column the output produced by the statement in the left-hand column. If the statement causes an error, fill in the right-hand column with either the phrase “compiler error” or “runtime error” to indicate when the error would be detected.

Statement Output

6. var1.method1(); \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

7. var2.method3(); \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

8. var1.method2(); \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

9. ((B)var1).method2(); \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

10. ((D)var5).method1(); \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

11.

Write a method, removeEWords, that accepts an ArrayList of Strings as a parameter. It should remove all of the words in the given list that contain the letter "e". Return the modified ArrayList. (4 points)

Example input/output:

{"roast", "toast", "beef"} --> {"roast", "toast"}

12. A Student class maintains an ArrayList of Grades. The Grade class is defined as:

The Student class is defined as follows:

a) Write the 1 argument constructor for the Student class. The constructor needs to initialize the ArrayList grades and the name field. (2 points)

public Student (String name) {

b) Write the Student class’s method getAverage(). Traverse the ArrayList of grades and return the average grade, a double. Compute the average by summing up all the Grade scores and totals, dividing the sum of the scores by the sum of the totals, and then multiply by 100.

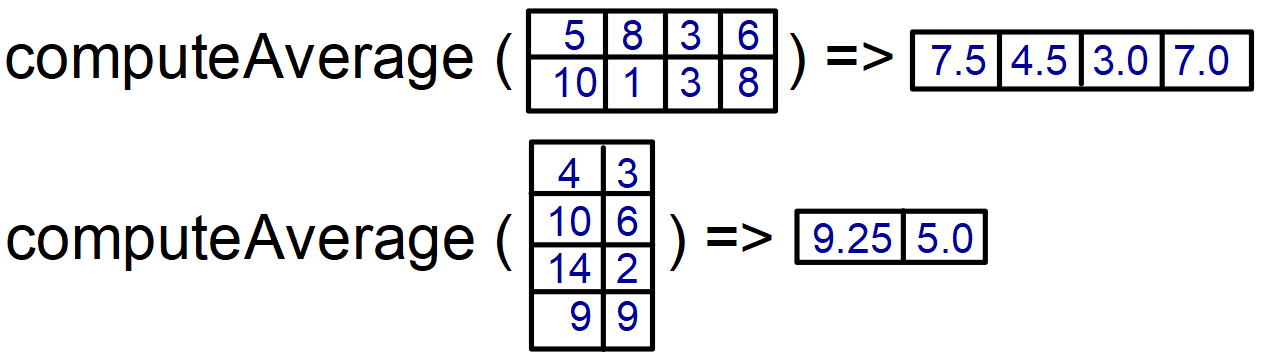
For example, if the list of grades are: 25/30 and 10/10, getAverage() would return (35/40) \* 100 or 87.5. If a Student has no grades, return -1. (4 points)

public double getAverage() {

c) The method, removeLowGrade() will traverse the ArrayList of grades and remove the grade with the lowest percentage (score/total). If two or more have the same lowest percentage, it does not matter which Grade is removed. A reference to the removed Grade object will be returned by the method. If the Student has no grades, return null. (5 points)

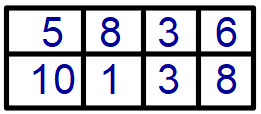
public Grade removeLowGrade () {

13. Write a method, computeAverage, that is given a 2-dimensional array of integers and computes the average value for each column. The average values are stored in a 1-dimensional array and returned by the method. You can assume the 2-dimensional array is not a jagged array (all rows have the same number of columns). ( 5 points)

**Example Input and Output:** 

14. Write a method, create2Darray, that takes in a 1-dimensional array and converts it into a 2 dimensional array. The method will also be passed the numbers of rows and columns. For example, if you are give a array with the following values: { 1, 2, 3, 4, 5, 6 } and rows = 3 and columns = 2, then you would return the 2D array: { {1, 2}, {3, 4}, {5, 6} }.

You can assume the product of the #rows and cols is equal to the length of the 1-dimensional array. (5 points)

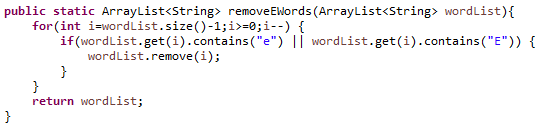
**Example**

create2Darray ( {5, 8, 3, 6, 10, 1, 3, 8}, 2, 4 ) =>

Answer Key:

1. B
2. D
3. E
4. D
5. A
6. A1
7. B3 A3
8. Compile Error
9. Compile Error
10. Run time error

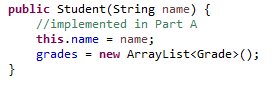
11.



4 points total

* 1 point - correctly defines the header
* 1 point - correctly uses for-loop correctly to remove items (either starts from the back or manually updates index)
* 1 point - correctly checks for both upper and lower case E
* 1 point - correctly returns the modified list (not a new list)

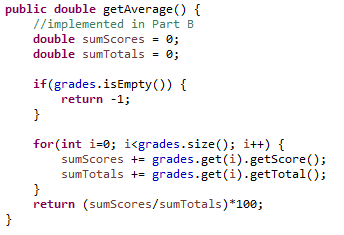
12a.



2 points total

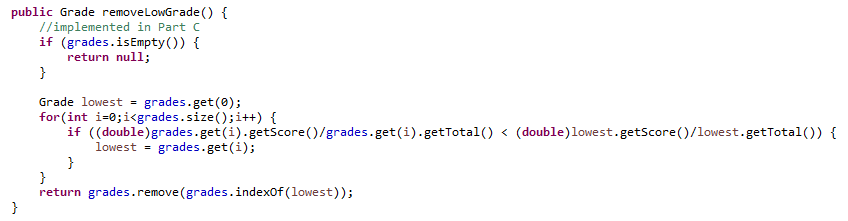
* 1 point - correctly initializes name
* 1 point - correctly initializes the ArrayList (using the constructor)

12b.



4 points total

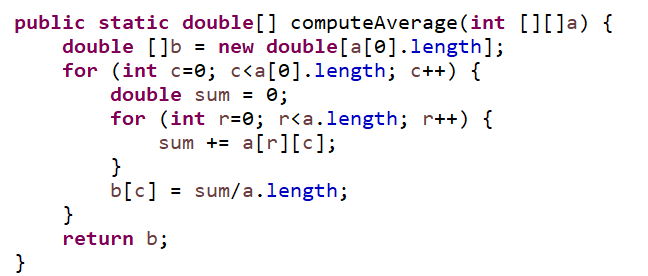
* 1 point - correctly returns -1 if there are no grades (the ArrayList is empty)
* 1 point - correctly uses a for-loop to traverse the ArrayList
* 1 point - correctly uses getters to get the score and totals
* 1 point - correctly does math to return the average as a percentage (not decimal)

12c.

5 points total

* 1 point - correctly returns null if there are no grades (the ArrayList is empty)
* 1 point - correctly uses a for-loop to traverse the entire ArrayList
* 1 point - correctly computes the averages
* 1 point - correctly finds the lowest grade
* 1 point - correctly removes and returns the lowest grade

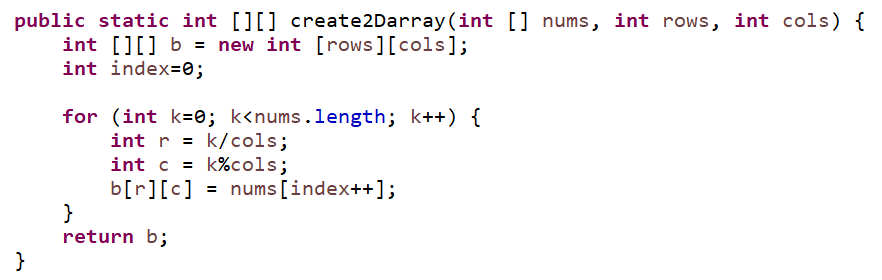
13.



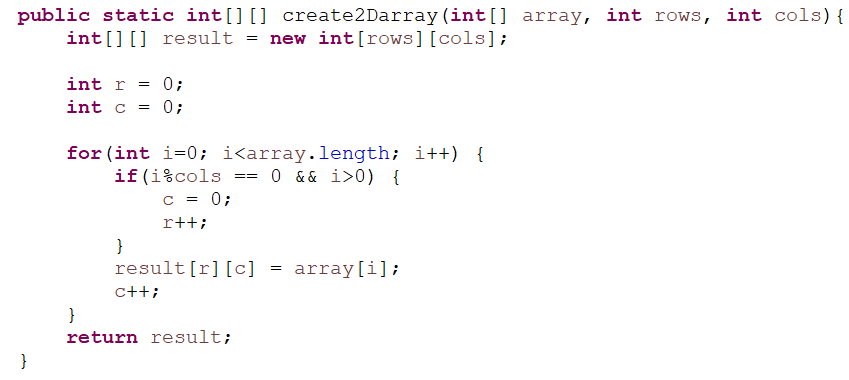
5 points total

* 1 point - correctly defines the header
* 0.5 point - correctly creates the 1-D array with the correct length
* 1 point - correctly uses a nested for-loop to traverse the 2-D array (0.5 points if there is a nested for-loop, full credit if the indices are correct (column-major))
* 1 point - correctly computes the average (must be a double)
* 0.5 point - correctly stores the average in the correct place in the 1-D array
* 1 point - correctly returns the 1-D array

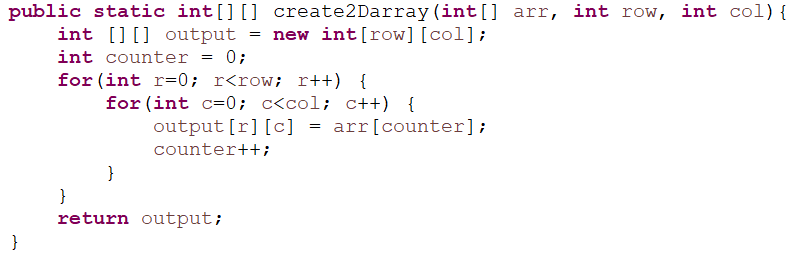
14.



An alternate solution:



And yet one more:



5 points total

* 1 point - correctly defines the header
* 1 point - correctly creates the 2-D array with the correct dimensions
* 1 point - correctly traverses the 1-D array to get elements
* 1 point - correctly places elements in the correct position in the 2-D array
* 1 point - correctly returns the 2-D array