

Name(s): \_\_\_\_\_

1. Analyze the code below and write the exact output of this program if we run it.

```
import java.util.ArrayList;
public class W6Activity1 {
    public static void main(String[] args){
        ArrayList<String> list_Strings = new ArrayList<>();
        list_Strings.add("Red");
        list_Strings.add("Green");
        list_Strings.add("Orange");
        list_Strings.add("White");
        list_Strings.add("Black");
        System.out.println(list_Strings);
        list_Strings.remove(1);
        list_Strings.remove("White");
        list_Strings.add(2, "Pink");
        for(int i = 0; i < list_Strings.size(); i++){
            System.out.println("Index: " + i + " Element: " + list_Strings.get(i));
        }
    }
}
```

[Red, Green, Orange, White, Black]

Index: 0 Element: Red

Index: 1 Element: Orange

Index: 2 Element: Pink

Index: 3 Element: Black

2. Identify instance variables and explain the methods in class W6Activity2.

```
import java.util.ArrayList;
import java.util.Scanner;
public class W6Activity2 {
    private ArrayList<Double> lst; //instance variable
    private int capacity; //instance variable
    public W6Activity2(int capacity){ //constructor - make sure that capacity is not
        if(capacity < 10) capacity = 10; //less than 10
        lst = new ArrayList<>(capacity); //creates lst ArrayList of Double with capacity
        this.capacity = capacity; //sets instance variable capacity
    }
    public boolean readAndAdd(Scanner in){
        System.out.println("Enter the quantity of numbers to read");
        int quantity = in.nextInt();
        if(quantity < (lst.size() + capacity)){ //if there is space to add in the lst
            for(int i = 0; i < quantity; i++){ //reads quantity of numbers and add each
                System.out.println("Enter a double number:"); //number to the lst
                double num = in.nextDouble();
                lst.add(num);
            }
            return true; //return true if was able to add numbers
        }
        return false; //return false if wasn't able to add numbers
    }
    public int size(){
        return lst.size(); //return the size of the lst
    }
    public int getCapacity(){
        return capacity; //return the capacity of lst
    }
    public String toString(){ //returns a String containing each number separated
        String msg = ""; //by a space and a new line at the end of the String
    }
}
```

## CS163/164: Worksheet Week 6 – ArrayLists

Name(s): \_\_\_\_\_

```

    for(Double element: lst){
        msg += element + " ";
    }
    msg += "\n";
    return msg;
}
}

```

3. Analyze the code below and write the exact output of this program if we run it with the following entries for the readAndAdd method: 3, 1.5, 2.4, 6.

```

import java.util.Scanner;
public class AppW6Activity2 {
    public static void main(String args[]){
        Scanner in = new Scanner(System.in); //creates a Scanner object in
        W6Activity2 w6 = new W6Activity2(4); //creates an object w6
        System.out.println(w6.size()); //prints the size of the lst
        System.out.println(w6.getCapacity()); //print the capacity of lst
        if(w6.readAndAdd(in)) //calls readAndAdd - assuming the numbers 3,1.5,2.4,6
            System.out.println("Number successful added!");
        else System.out.println("Not possible to insert more numbers!");
        System.out.println(w6.size()); //prints size of the lst
        System.out.println(w6.getCapacity()); //print capacity
        System.out.println(w6.toString()); //print 3 1.5 2.4 6
    }
}

```

4. Write the following methods for the W6Activity2 class:
- A method that finds and returns the maximum value stored in the ArrayList.
  - A method that finds and return the index where the minimum value is stored in the ArrayList.
  - A method that returns the sum of all elements in the ArrayList.
  - A method that returns the average of all elements in the ArrayList.
  - A method that returns a list with elements that are less than the average of all elements in the ArrayList.

Before you code those methods think about how you are going to solve each one of those problems. Write your algorithm in English first, then translate your algorithm to a Java method.

```

import java.util.ArrayList;
import java.util.Scanner;
public class W6Activity2 {
    private ArrayList<Double> lst;
    private int capacity;
    public W6Activity2(int capacity){
        if(capacity < 10) capacity = 10;
        lst = new ArrayList<>(capacity);
        this.capacity = capacity;
    }
    public boolean readAndAdd(Scanner in){
        System.out.println("Enter the quantity of numbers to read");
        int quantity = in.nextInt();
        if(quantity < (lst.size() + capacity)){
            for(int i = 0; i < quantity; i++){
                System.out.println("Enter a double number:");
                double num = in.nextDouble();
                lst.add(num);
            }
            return true;
        }
    }
}

```

## CS163/164: Worksheet Week 6 – ArrayLists

Name(s): \_\_\_\_\_

```

        }
        return false;
    }
    public int size(){
        return lst.size();
    }
    public int getCapacity(){
        return capacity;
    }
    public String toString(){
        String msg = "";
        for(Double element: lst){
            msg += element + " ";
        }
        msg += "\n";
        return msg;
    }

    public double maxValue(){
        double max = lst.get(0);
        for(int i = 1; i < lst.size(); i++){
            double num = lst.get(i);
            if(num > max)
                max = num;
        }
        return max;
    }

    public int indexMinValue(){
        double min = lst.get(0);
        int index = 0;
        for(int i = 1; i < lst.size(); i++){
            double num = lst.get(i);
            if(num < min){
                min = num;
                index = i;
            }
        }
        return index;
    }
    public double average(){
        double sum = 0;
        for(Double element: lst){
            sum += element;
        }
        return sum/lst.size();
    }
}

```

5. In the main method of class App W6Activity2, write one method call for each method you created previously in question 4.

```

import java.util.Scanner;
public class AppW6Activity2 {
    public static void main(String args[]){
        Scanner in = new Scanner(System.in);
        W6Activity2 w6 = new W6Activity2(4);
        System.out.println(w6.size());
        System.out.println(w6.getCapacity());
        if(w6.readAndAdd(in))

```

## CS163/164: Worksheet Week 6 – ArrayLists

Name(s): \_\_\_\_\_

```
        System.out.println("Number successful added!");
    else System.out.println("Not possible to insert more numbers!");
    System.out.println(w6.size());
    System.out.println(w6.getCapacity());
    System.out.println(w6.toString());
    if(w6.size() > 0){
        System.out.println(w6.maxValue());
        System.out.printf("Index of min value %d\n", w6.indexMinValue());
        System.out.println("Average: " + w6.average());
    }
}
```