**CSC142, Computer Science II, Project 3 assignment**

Answer all the questions and submit both document and java files to D2L. Later submission is not accepted.

**1. Understand the code before you verify the result with the computer.**

Draw a representation of what the computer's memory (of each variable) looks like at the end of each of these programs:

public class Array-Declarations {

public static void main(String [] args) {

int [] w;

int [] x = null;

int [] y = new int[3];

int [] z = {1, 3, 5, 7, -14};

}

z

?

w

x

y

}

null

length

3

|  |  |  |
| --- | --- | --- |
| 0 | 0 | 0 |

length

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 3 | 5 | 7 | -14 |

5

*On an exam or quiz, it's fine to leave out the box for the length, since it's usually obvious what the length is. In the examples below, I will leave out the box for the length. But just so you know, this is the full representation of what memory looks like.*

public class Array-Assignment {

public static void main(String [] args) {

int [] x = new int[3];

int [] y = {1, 3, 5, 9};

x[2] = y[3];

x[0]++;

y[1] += y[2] - y[0];

int [] z = x;

x = y;

x[1] = 4;

}

}

x, y, z?

public class Array-Length {

public static void main(String [] args) {

int [] x = new int[0];

int [] y = {};

int [] z = {0};

}

}

x, y, z?

public class Array-With-Loop1 {

public static void main(String [] args) {

int [] x = {-4, 3, 8, 2, -5, 1};

for(int i=1; i<x.length; i++) {

x[i] += x[i-1]; // notice: += instead of =

}

}

}

x?

public class Array-With-Loop2 {

public static void main(String [] args) {

int [] x = {-4, 3, 8, 2, -5, 1};

for(int i=1; i<x.length; i++) {

x[i] = x[i-1];

}

}

}

x?

**2. Write a counter controlled loop to solve the following problems. Each one will involve an array (see the sample of Weather.java on slide 20).**

1. MinMax.java

Read in 25 ints from the keyboard, and store them in an array, then

find the maximum and minimum values in such an array and display them on screen.

Source code?

1. Swap.java

Read in 25 ints from the keyboard, and store them in an array “myArray”, then

find the *position* (or *index*) of the maximum and minimum values in the array, and *swap* them (i.e., move the biggest element to the position of the smallest, and move the smallest element to the position of the biggest).

At the end, apply

System.out.println(Arrays.toString(myArray))

to print out the new status of such an array “myArray”. If necessary, include

import java.util.\*;

or

import java.util.Arrays;

Source code?

1. Even.java

Read 25 ints from the keyboard, and store them in an array.

Display "true" on the screen if there is an even number of even numbers among these. Otherwise, display "false".

Source code?