CST8116 Lab Exercise 09

Array with Elements of Primitive Type

# Instructions Part 1: Pseudocode, Flowchart

* A client wants a program that permits them to enter 5 numbers with decimal points, then display the five numbers on screen in the reverse order, each input must be labeled as Input Number 1, Input Number 2, etc. and the output as Number 5 is x, Number 4 is x, where x represents the value stored in the array.
* To keep this exercise light in work, there is no requirement to validate the user input, assume it will be entered correctly.
* You are required to use an array to store the values as they are input using a loop.
* Displaying the values in the reverse order should be accomplished using a second loop.
* You may use any form of loop of your choosing (while, do-while, or for).
* Create the pseudocode and flowchart for the problem and include these in your MS Word document, note that your full name must appear within any flowcharts you submit.
* Hand drawn flowcharts are acceptable as images within the MS Word document, computer generated flowcharts can also be included in your MS Word document as an image (or MS Word shapes themselves are acceptable).

# Instructions Part 2: Testing / Hand Trace

* Provide a hand-trace of your first loop for entering values, including the loop control variable, user input, and array index assignment.
* Provide a second hand-trace of your second loop for outputting the values in the reverse order
* This is an example, you will have more steps in your traces. Feel free to use these as a starting point.
* Depending on how you decide to do your logic your loop control variable does not need to start with a value of 1.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Loop control variable | Should Continue | Output | Input Value | Array Index Value |
| 1 | Y | Enter Number 1 | 10.10 | 0 |
|  |  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Loop control variable | Should Continue | Output | Array Index Value |
| 5 | Y | Number 5 is 50.50 | 4 |
|  |  |  |  |

# Instructions: Part 3 Plan, Create, Compile, Run a Java Program

* Using Eclipse, create a project named Exercise09 within your CST8116Workspace folder. (See Hybrid 1)
* Based on your pseudocode and flowchart, write the Java program to perform the tasks required.
* Follow Java programming conventions for identifiers, indentation, and provide programmer comments at the top of your file, and on each class header, and constructor and method.
* Take a screen shot of your program running, using the inputs for the planned hand-trace you created for entry of five numbers and the output of the numbers in reverse.
* Ensure your full name is visible as part of the program output in your screen shot.

# Instructions: Part 4 Memory Map

* Refer to Hybrid 09
* Draw a memory map for your array in memory just after it has been filled with the values entered by the user (use the values in your hand-trace table).
* Start using the identifier you used in code for the array (your variable name for the array).

# Microsoft Word Document Format

See the template example (from exercise 01), suggested headings below:

Pseudocode

Flowchart

Hand Trace

Memory Map

Java Screen Shot

**Note: You are not required to copy and paste your Java code into the MS Word document, however you must upload the .java file(s) in addition to your MS Word document.**

# Submission Requirements

* Upload your MS Word document as well as your Java file to the Brightspace submission area by the due date. (See Brightspace for due date).
* Follow any additional submission requirements specified by your lab professor when submitting your homework.

# Sample Program Run (User inputs highlighted in yellow, bold font)

Enter Number 1 **10.10**

Enter Number 2 **20.20**

Enter Number 3 **30.30**

Enter Number 4 **40.40**

Enter Number 5 **50.50**

Number 5 is 50.5

Number 4 is 40.4

Number 3 is 30.3

Number 2 is 20.2

Number 1 is 10.1

Program by Stanley Pieda

# Grading (6 Points)

|  |  |  |  |
| --- | --- | --- | --- |
| Criteria | Missing / Poor (0) | Below Expectations (0.5) | Meets Expectations (1) |
| Algorithm: pseudocode | Missing or poorly done. | Pseudocode is included in MS Word document, however there are logic mistakes and or missing either loops and / or an array | Pseudocode is included in MS Word document, provides a working algorithm that meets the requirements. |
| Algorithm: flowchart | Missing or poorly done or student name missing from flowchart. | Flowchart is included in MS Word document, however there are logic mistakes. | Flowchart is included in MS Word document, has correct format, and provides a working algorithm that meets the requirements. |
| Hand Trace | Missing or poorly done | Hand Trace does not show entry of two sets of student grades. | Test plan does test the minimal requested items, and also adds tests so that each possible letter grade is tested. |
| Screen shots running program | Missing, or missing students name as part of the program output. | Screen shots show program execution with sample input that matches the hand trace. May not show both a single student-grade-set entry, and a two student-grade-set entry runs of the program. | Screen shots show program execution with sample input that matches the hand trace. Shows both a single student-grade-set entry, and a two student-grade-set entry runs of the program. |
| Java file(s): Comments and logic | No programmer comments, and or no student name. | Not all parts are commented: Comment header block at top of each source code file, class and constructor(s), and method(s) each have brief comment.  Small syntax or logic mistakes in the program. Java coding conventions not fully followed. Program accepts inputs, provides outputs, outputs may not be fully correct. | All parts are commented: Comment header block at top of each source code file, class and constructor(s), and method(s) each have brief comment.  Program syntax and logic are correct. Java coding conventions fully followed. Program accepts inputs, provides correct outputs. |
| Memory Map | Missing, or poorly done. | Partly correct, may not have all references mapped correctly. | Correct, shows reference to array, references to Employee objects, with references to Strings for name. |