GUI Programming 2019-2020 – Year 2

Labwork 1 - Java Revision 1 (Arrays):

(Worth 5% - or 50 points out of 500 points for labwork this semester)

IMPORTANT NOTES:

- **NO COPYING PERMITTED AND ZERO MARKS WILL APPLY TO COPIED WORK. FURTHER ACTION MAY BE TAKEN AGAINST STUDENTS THAT HAVE BEEN FOUND TO COPY WORK.**

- **ASSESSMENT WILL INVOLVE ONE-TO-ONE QUESTIONS ABOUT YOUR SUBMITTED WORK. USE COMMENTS IN YOUR CODE TO ENSURE YOU DON'T FORGET WHY YOU WROTE CODE YOU MAY LATER BE ASKED ABOUT.**

- **ALL WORK MUST BE SUBMITTED TO MOODLE BY DATES SPECIFIED (2 LABS SUBMISSIONS OF FIVE LABS THROUGHOUT THE SEMESTER).**

- **MANY OF THE TASKS ASSIGNED BELOW CAN BE COMPLEX AND\OR THE DESCRIPTIONS MAY REQUIRE FURTHER CLARIFICATIONS. PLEASE USE THE AVAILABLE LAB TIMES TO ASK FOR CLARIFICATIONS AND ADVICE\HINTS ON THE TASKS BELOW.**

Part 1 – Basic Array – type char (5 points)

Create a Java program that defines an array of characters (type **char**). Make the array large enough to store your FULL NAME (include middle names etc.). Store your full name **in characters only (not Strings!!!)** into the array.

Required activities and marking guideline:

- Create the array (2 points)
- Set array size (1 point)
- Fill the array with correct data (2 points)

Part 2 – Basic Array – type int (5 points)

Create a Java program that defines an array of integers (type **int**) to store your date of birth in numeric form. Make the array long enough to store your date of birth in the following format (no leading zeros): 1 1 1901.

Required activities and marking guideline:

- Create the array (2 points)
- Set array size for your date of birth (1 point)
- Fill the array with correct data (2 points)

Part 3 – Basic array and traversing array (7 points)

Create and array of type **double** to store the amount of predicted rainfall exactly where you live in the next 10 days (use millimeters of rainfall). You can use a weather website or app for the data (include a link to the website you got the data from in the comments of the Java program). Print out the data using a loop,
e.g.,   Day 1: 10.5mm
        Day 2: 0.5mm etc.

Required activities and marking guideline:

- Create the array of type double   (2 points)
- Set size of the array   (1 point)
- Fill the array with rainfall data   (2 points)
- Use a loop to output data in readable\neat format   (2 points)

Part 4 – Two dimensional array (13 points)

Create a two-dimensional array that shows a grid of the rainfall, temperature and wind speed where you live in the next 10 days. The output should look similar to the following (you may re-use the same rainfall data from Part 3):

    Day 1: 10.1 mm   10.5 C   10.3 knts
    Day 2: 0.5 mm   12.5 C   5.5 knts

Required activities and marking guideline:

- Create the array of arrays (two-dimensional)   (4 points)
- Fill the array with all data   (4 points)
- Use a loop to output data in readable\neat format   (5 points)

Part 5 - Array processing – alphabet shift encoder (20 points)

Using ONLY arrays create an **alphabet shift** coder and decoder system in Java. The system should be able to output information in coded form and also output information in decoded. Use your name and a favorite place of yours as a code and decode example to prove your system works. Messages to code must be written using an alphabet coder array, e.g., array[0] is A, so message ABC is written {0,1,2} or {array[0],array[1],array[2]}. The amount of alphabet shift on the coder system must be changeable using a **final static variable** (aka "a constant"), e.g., if the shift coder variable is set to +2 then A becomes C (two more than the original message) and if the shift variable is set to -2 then A becomes X (two letters less than A). Example: If the word "CODE" is coded at shift value of +2 then the coded equivalent would be "EQFG". In order to simplify the system you may assume the use of ALL CAPS for this alphabet shift encoding system. The coded messages can be hard-coded (no need to input from keyboard). Add comments to your code where you carry out a complicated task so that you can remember what you did when it comes to assessment!!!

Required activities and marking guideline:

- Create alpha shift array A -> Z   (3 points)
- Create coded message array using alpha array   (3 points)
- Set final static variable for alphabet shift   (2 points)
- Test coded message   (3 points)
- Deals with shift variable of less than zero   (3 points)
- Deals with shift variable greater than maximum   (3 points)
- Uses loops for iterating the array(s)   (3 points)
- Fully working to spec   (1 point)