CPSC 1061 – Introduction to Programming in Java Lab

Spring 2021

Lab 8 – Due Monday, March 8, 10:00pm

1 Introduction and Lab Objectives

In this lab you will practice using arrays in Java programs. The objectives of this lab are to:

1. declare, create, and initialize arrays
2. access array elements
3. program common array operations
4. use arrays in methods
5. continue using previous concepts such as creating random numbers, working with Strings, getting user input, and using if-statements and for-loops

The lab today **can** be performed in groups of two but does not have to. If you work in teams, do not just tell each other solutions but always make sure that your lab partner also understands why something does or does not work. Have fun!

2 Main

2.1 General Instructions

At the start of each program, write your name, the name of your lab partner (if you chose to not have a lab partner, write "no lab partner", the field still needs to be there), the course and lab, the date, and a description of what your program does as in the previous lab. In this lab as well as in all following labs, each program needs to have comments (not just at the beginning), to be clean, and to compile. Furthermore, any input and output should be designed to have appropriate instructions and sentences.

2.2 Fun.java

Write a program to do the following:

1. Create an array to hold 3 values of type char
2. Assign the character 'f' to the first element
3. Assign the character 'u' to the second element

4. Assign the character 'n' to the last element
5. Write a loop that prints all the elements of the array (no spaces or line breaks)

2.3 OldMacDonald.java

Write a program to print the lyrics of the song "Old MacDonald Had a Farm":

1. Use an array initializer to create, declare, and initialize an array to hold the values "cow", "dog", "cat", "horse", "lamb", and "chicken" in one step
2. In a similar way, create a second array to hold the values "moo", "woof", "meow", "neigh", "baa", and "cluck"
3. Create a constant with the value "E-I-E-I-O"
4. Write a loop that prints the lyrics using the arrays and the constant, see beginning of lyrics below

```
Old MacDonald had a farm
E-I-E-I-O
And on his farm he had a cow
E-I-E-I-O
With a moo moo here
And a moo moo there
Here a moo, there a moo
Everywhere a moo moo

Old MacDonald had a farm
E-I-E-I-O
And on his farm he had a dog
E-I-E-I-O
With a woof woof here
And a woof woof there
Here a woof, there a woof
Everywhere a woof woof

Old MacDonald had a farm
E-I-E-I-O
And on his farm he had a cat
E-I-E-I-O
```

2.4 BooleanArray.java

Write a program to do the following:

1. Create an array to hold 10 values of type boolean
2. Write a loop that assigns true or false randomly to each element
3. Write a loop that counts how many times true occurs
4. Print all the elements of the array as well as your result from counting the number of times the value true occurs

2.5 PrintArray.java

Write a method that takes an array of numbers (type double) as a parameter and prints the array in a nicely formatted way with spaces between each element of the array and brackets at the beginning and end.

Create an array with five numbers and ask the user for numbers to initialize the array. Invoke your method to display the array.

2.6 AddVectors.java

In this program we are using arrays to represent mathematical vectors. Write a method addV that takes two arrays of numbers (two vectors) of the same length and returns a new array containing the sums of the corresponding elements of each.

If the method receives two vectors of different lengths it should print an error message and return an array of length 1 with the value 0.

In your main method write a test program to test the method. First define the following vectors:
$\vec{u_1} = [1, 2, 3, 4]$
$\vec{v_1} = [-5, -6, -7, -8]$
$\vec{u_2} = [4, 8, 15]$
$v2 = [16, 23, 42]$

Print the results of the following operations:
$\vec{u_1} + \vec{v_1}$
$\vec{u_1} + \vec{u_2}$
$\vec{v_1} + \vec{v_2}$
You can reuse the method from the previous program.

2.7 Submit Files

Make sure to test your java files on the lab machines. Create a single zip-file that includes all the java files (and no other files) and submit the zip-file to Canvas.