

Tutorial 4: Arrays & Debugging

Aim:

- Get familiar with Java arrays and debugging.
- Consolidate learning from week 3, 4.
- Get feedback.

Note: Use the material from the lecture if you need. If you find any problem or have a question, ask your tutor. If you do not have enough time during the session, it is recommended that you finish the exercises at home. Challenges are optional, although recommended.

Section 01: Main Questions

Q1: Arrays creation, access, and update.

Write a program to do the following.

- a. Create an array of integers with size 06 .
- b. Prompt the user to enter 6 marks and store them in the array. (Read user input)
- c. Then, display the 10 values in the array **using enhanced for loop**.
- d. Calculate the number of students who have failed (mark lower than 40) and the average mark.
- e. Print the number of students who have failed and the average.

Q2: Arrays: swap values I.

Write a program in Java that, given two indices and the following array, swaps their values:

```
String[] array = {"Alex", "Max", "Charlie", "Bob", "Ada", "Jim"};
```

Example: if we swap index 2 and index 4, the result should be:

Alex, Max, **Ada**, Bob, **Charlie**, Jim

Q3: Arrays: swap values II.

Write a program where an array of length 10 is filled with double values. The user then inputs 2 integer values i and j and the values in the array position i and j are swapped around(i and j can be any valid index number of the array). The program should then print out the contents of the array. What happens if i or j are larger than 10?

Q4: Search.

Given the set (array) A = {2, 3, 4, 5, 6, 7, 8, 9} find all elements of set A that are:

- a) Even numbers
- b) Those numbers that being multiplied by 2 give a number that is also an element of A.

Write a Java program that calculates and prints the solution of the previous two tasks.

Note: To find if a number is even or odd, you can calculate the remainder of a division by using the operator %. An even number is a number that is divisible by 2 and generates a reminder of 0.

Acknowledgement: This exercise was taken from Mathematics in Computing (4COSC002W).

Q5: Multi-dimensional arrays

You have 5 students, and each student has 3 marks from 3 different subjects (i.e: SD2,Web,Maths).

Write a Java program that asks for the 3 marks of the 5 students and finds the average of each student.

Use the following multidimensional array to store the marks:

```
double[][] marks = new double[5][3];
```

Q6: Search

You are given a sorted array of integers which contains runs scored by a batsman in 10 innings (index of the array represents the match number). Find out in which match, batsman has scored 47 runs. (Use binary search to implement the solution).

```
int [] runs = {05,12, 28, 34, 47, 52, 65, 71, 80, 99}
```

Q7: Debugging: Find the bugs/errors.

We have the following Java program:

```
public class Tutorial3_ex4 {  
    public static void main(String[] args) {  
        for(int i = 9; i > 0; i--){  
            for(int j = 9; j > 0; j--){  
                System.out.print(j);  
            }  
            System.out.println();  
        }  
    }  
}
```

That prints the following output:

```
987654321  
987654321  
987654321  
987654321  
987654321  
987654321  
987654321  
987654321  
987654321  
987654321
```

However, this output is not correct and instead, should print:

```
999999999
888888888
777777777
666666666
555555555
444444444
333333333
222222222
111111111
```

The code has a bug and therefore the output is not the expected. Use the rubber duck debugging method to find the bug. (Refer lecture notes)

Section 02: Challenging Questions

Exercise 06: Probability.

A fair die is rolled twice and we get two numbers X = result of the first roll and Y = result of the second roll.

- a) What is the probability that $X = 4$?
- b) What is the probability that $Y = 4$?
- c) What is the probability that both $X = 4$ and $Y = 4$?

Write a program in Java that simulates this scenario 1,000 times (you roll two dice 1,000 times) and estimates the probabilities. How good is your simulation? Is it any better if you increase the number of simulations to 1,000,000?

Note: You can use the Java class `Random` to generate random numbers (you will need to import `java.util.Random`. You can then create a generator: `Random generator = new Random();`

*Acknowledgement: This exercise was taken from *Mathematics in Computing* (4COSC002W).*

Q7: Mark adjustment

Write a program that reads in and stores 10 student marks and then reduces each mark by 10% (because it was 24 hours late). Use a loop to reduce each mark. Each reduced mark is then stored in the same element of the array. You should then output the contents of the array.

Q8: Shift Elements

Write a program that reads in a sequence of ints into an array. The program should then shift each value on one position in the array. For example, whatever is in array position 4 should be moved into position 5. Whatever is in the last should be moved to the start. The program should then print out the contents of the array.

Q9: Sorting elements

Write a program to get 10 Integers from the user and store them in an array called `myNamesArray`. then clone the Integers in the `myNames` into another array called `myNamesArray2`. Sort the `myNamesArrayList` using one of the sort methods available in lecture note.

Q10: Pass by value vs pass by reference

Are arrays passed by reference or passed by value? Justify your answer with an example code snippet

Q11: Pass by value vs pass by reference

Write a method `removeBadPairs` that accepts an Array of integers and removes any adjacent pair of integers in the list if the left element of the pair is larger than the right element of the pair. Every pair's left element is an even-numbered index in the list, and every pair's right element is an odd index in the list.

Q12: Remove bad pairs

Suppose an array called `list` stores the following element values: `[3, 7, 9, 2, 5, 5, 8, 5, 6, 3, 4, 7, 3, 1]`

We can think of this list as a sequence of pairs: (3, 7), (9, 2), (5, 5), (8, 5), (6, 3), (4, 7), (3, 1). The pairs (9, 2), (8, 5), (6, 3), and (3, 1) are "bad" because the left element is larger than the right one, so these pairs should be removed. So the call of `removeBadPairs(list)`; would change the list to store the following element values: `[3, 7, 5, 5, 4, 7]` If the list has an odd length, the last element is not part of a pair and is also considered "bad;" it should therefore be removed by your method. If an empty list is passed in, the list should still be empty at the end of the call. You may assume that the list passed is not null. You may not use any other arrays, lists, or other data structures to help you solve this problem, though you can create as many simple variables as you like.

Section 03 : HackerRank Challenges

HackerRank: Interview questions. Solve the following tasks in HackerRank:

1. Java 1D Array
2. Java 2D Array
3. Java ArrayList
4. Java 1D Array (Part 2) [more challenging]