McGill

School of Continuing Studies École d'éducation permanente

CCCS 315: Data Structures and Algorithms

Assignment #4

Type of Assignment

- *Individual Work*
- Evaluated*: this assignment is marked out of 100 points and is worth 10% of your final grade*
- Estimated Time*: 120 minutes*

Description

In this Assignment you will use algorithms and tools you learn to sort an algorithm.

Learning outcomes being met through this assessment

- Apply Big-O notation to analysis an algorithm
- Apply knowledge of general sorting problems
- Use the best algorithm to implement

Steps to complete the assignment

This assignment is designed like a job interview style. First, we will introduce a problem. You then need to apply critical thinking, design a solution, evaluate the solution (Big-O), then improve it if needed. Lastly, you will implement your solution.

We will create a list of random integers like assignment 2, create 3 methods of sorting them then sort them. This time, you choose 3 algorithms but not the one we coded in A2.

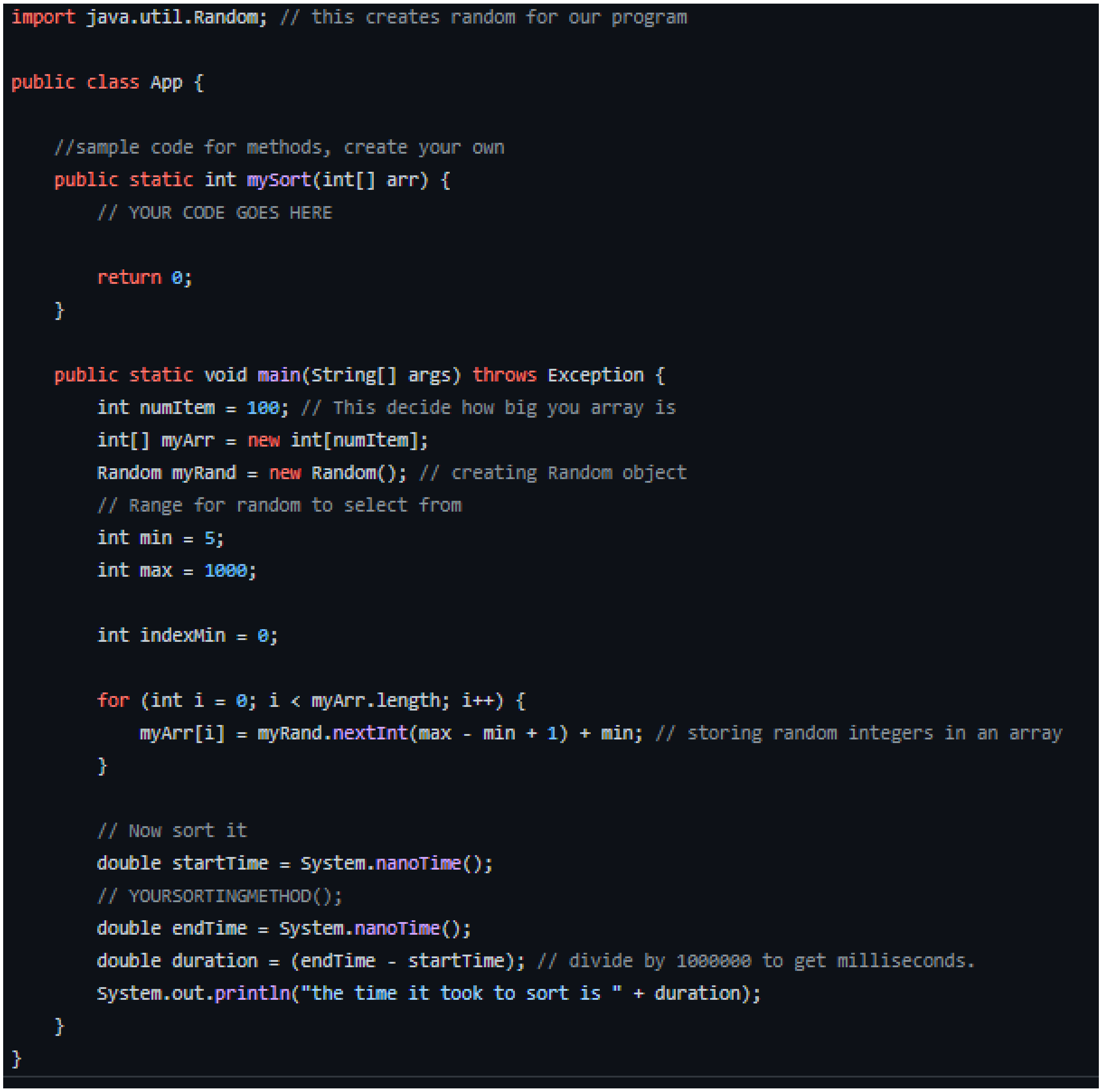
Part 1: Stacks

- No code
- Pseudocode: Write 3 algorithm for sorting (Not used before!!!)
- Calculate their time complexity (as a function of f(n))
- Then calculate their big-O
- Decide which one is better when our input size n is 100 vs 10 vs 1000

Part2: LinkedList insert

We will implement a program that will use your sorting algorithm. We will create a list of random integers, create 3 methods of sorting them then sort them.

- Use the following code:



```
import java.util.Random; // this creates random for our program

public class App {

    //sample code for methods, create your own
    public static int mySort(int[] arr) {
        // YOUR CODE GOES HERE

        return 0;
    }

    public static void main(String[] args) throws Exception {
        int numItem = 100; // This decide how big you array is
        int[] myArr = new int[numItem];
        Random myRand = new Random(); // creating Random object
        // Range for random to select from
        int min = 5;
        int max = 1000;

        int indexMin = 0;

        for (int i = 0; i < myArr.length; i++) {
            myArr[i] = myRand.nextInt(max - min + 1) + min; // storing random integers in an array
        }

        // Now sort it
        double startTime = System.nanoTime();
        // YOURSORTINGMETHOD();
        double endTime = System.nanoTime();
        double duration = (endTime - startTime); // divide by 1000000 to get milliseconds.
        System.out.println("the time it took to sort is " + duration);
    }
}
```

Code can be downloaded in our github:

From https://github.com/farhadrclass/CCCS315/blob/main/A2/src/App.java
Remember you do not need to use it but it helps

- We provided the code for you to start
    - **Create your sorting algorithm**
    - **Run them with n is 10, 100, or 1000**
- %5 only: if you can run them many times and average the time of running, it would be much easier to see the behavior.
- Answer:
    - Can you comment which one is faster?

  - Why did you choose them? What was the big-O of them, did they perform the way you expected?
  - Can you re-do it in such a way that your array (your random array, if not random) is worst-case scenario?
  - **Submit it in PDF**

Evaluation Criteria

- Format Requirements (10%)
  - Comments and CamelCase is important
  - Style
- Part 1 (30%)
  - 5% : deciding the algorithm
  - 10%: Pseudocode
  - 15%: Big-o and function analysis
- Part 2(70%)
  - Classes 20%
  - Sorting algorithm 30%
  - Analysis: 20%

Submitting your Work

- You have to zip all the Java code for part 1 and part 2 (.java codes and if missing you will lose 85%)
- PDF for last part of part2

Notes:

  -
- Cheating or copying online without referencing correctly or explaining your code before and after online source has automatic failure as a grade.
  - If your references are not correct, you will lose marks.
  - (In Comments) Show your code before searching online AND showing what the issue was.
  - (In Comments) The reference.
  - (In Comments) How the online or book reference helped you change your code.
  - **Copying 80% from online with reference still results in ZERO!**
    - **Modify it to fit your code, do not blindly copy it and remember you are here to learn !**
- Do NOT share your code with anyone except your TA and me.
  - otherwise you will get a grade of ZERO.