Exercise 03: Analysis of Algorithms

In-Lab Exercise

Objectives

The aim of this exercise is to explain why do we classify the growth rate of the running time of an algorithm using the Big-O notation. For example if $T(n) = 2n^2 + 6n$, why do we have to `simplify` it to $T(n) = O(n^2)$? What's the point?

Tasks

For the programming part of this exercise, you're going to run the answer key to your Pre-Lab Exercise. It means that the `InLab` subfolder is similar to the the `PreLab/answers` folder.

Plese make sure that you have 30 minutes to an hour to solely use your computer to do this exercise. Try to minimize unnecessary fluctuations in the performance of your hardware.

You will run the program using the `make silent` command in your terminal. The `make silent` command will use `program2.cs` as input to the compiled `main2.c` program. Here are the default contents of `program2.cs`.

b 10000 d

b 20000 d

b 40000 d

b 80000 d

b 100000 d

b 200000 d

b 400000 d

b 800000 d

The following are the possible scenarios that you might encounter when you run this program at home:

1. You get zero elapsed time for all the runs. You need to contact your lab instructor ASAP when you encounter this.
2. It is taking you so long, the program has been for more than 30 minutes and it still yet finished. Terminate the program and delete the following lines from `program2.cs`:

b 100000 d

b 200000 d

b 400000 d

b 800000 d

Re-run your program and if you get non-zero elapsed time results this time, proceed to #3 scenario. Otherwise, contact your lab instructor.

3. You generated non-zero elapsed time results.

When you generated non-zero results of elapsed times, re-run your code using `make silent > results.txt`.

Submission

After successfuly running `make silent > results.txt`, rename `results.txt` to `<initials><surname>results.txt` (e.g.`ajjacildoresults.txt`). Submit the said file to our google classroom.

Questions?

If you have any questions, contact your lab instructor.