

CS 201, Spring 2019

Homework Assignment 2

Due: 23:55, April 16, 2019

In this homework, you will analyze and compare algorithmic complexity of four different solutions for the **Maximum Subsequence Sum Problem**. For that, first read Section 2.4.3 (Solutions for the Maximum Subsequence Sum Problem) from the handout and study each of these four solutions. Be sure that you understand how the upper bounds are found for the running time of each solution. Then, implement these solutions in C++. Note that you may use the codes provided in the handout.

Next, write a driver (main) function that generates an array that contains integers and calls each of these solutions with that array as input. Run each solution on a computer and record execution times when different input sizes are used. You are expected to try many different input sizes, both small inputs and very large inputs (as large as thousands to millions depending on the solution), and observe the effects of different growth rates.

In this assignment,

1. Use your results to generate a comparison table and a plot of running time (y-axis) versus the input size N (x-axis). In particular, you are expected to produce a table and a plot as in Figure 2.2 and Figure 2.3 of the handout, respectively. In the table, each row corresponds to an input size and each column holds the running times of a specific solution.
2. Plot the expected growth rates obtained from the theoretical analysis (as given for each solution) by using the same N values that you used in obtaining your results. Compare the expected growth rates and the obtained results, and discuss your observations in a paragraph.
3. Provide the specifications of the computer you used to obtain these execution times. You can use any computer with any operating system for this assignment.

You can use the following code segment to compute the execution time of a code block. For these operations, you must include the **chrono** header file.

```
// Declare necessary variables
std::chrono::time_point< std::chrono::system_clock > startTime;
std::chrono::duration< double, milli > elapsedTime;

// Store the starting time
startTime = std::chrono::system_clock::now();

// Code block
...

// Compute the number of milliseconds that passed since the starting time
elapsedTime = std::chrono::system_clock::now() - startTime;
cout << "Execution took " << elapsedTime.count() << " milliseconds." << endl;
```

NOTES ABOUT SUBMISSION:

1. This assignment is due by 23:55 on Tuesday, April 16, 2019. You should upload your homework to the upload link on Moodle before the deadline. This upload link will be available between April 5th and 19th. No hardcopy submission is needed. The standard rules about late homework submissions apply. Please see the course web page for further discussion of the late homework policy as well as academic integrity.
2. In this assignment, you must submit a report (as a pdf file) that contains all information requested above (plots, tables, computer specification, discussion) and a cpp file that contains the main function that you used as the driver in your simulations as well as the solution functions in a single archive file.
3. This homework will be graded by your TA Furkan Hüseyin (furkan.huseyin at bilkent edu tr). Thus, you may ask your homework related questions directly to him.

VERY IMPORTANT: We expect all of you to comply with academic integrity. The honor code statement, which has been sent to you by email, was prepared to clarify our expectations about the academic integrity principles. Please study the part of this statement related with “individual assignments” very carefully.

If you submit this homework assignment, we will assume that you all read this honor code statement and comprehensively understood its details. Thus, please make sure that you understood it. If you have any questions (any confusions) about the honor code statement, consult with the course instructors.

We will check your submissions for plagiarism.