CS 201 – Spring 2024

Homework 2: Algorithm Efficiency and Sorting

Due: April 8, 2024, 23:59

Introduction:

The Bilkent Campus is renowned for its cleanliness and aesthetic appeal among Turkish universities. There is a new campus extension that lacks trees, prompting the administration to plan a tree-planting initiative before spring. To ensure uniformity in the trees' height, the Computer Engineering Department is given the task of employing sorting algorithms. Despite challenges such as trees from different suppliers and varying truck sizes, sorting algorithms offer a promising solution to reduce labor hours and achieve a cohesive landscape.

Your task is to prepare a report based on the conditions below to give the Bilkent Administration insights about the best way of sorting their nascent trees.

Assumptions:

- Comparing the heights of a pair of trees takes 1 second.
- Changing the position of each tree takes 5 seconds (Swapping = Three position changes).
- For the simulations, we assume that the truck capacities are powers of 2 that range between $\mathbf{2^3}$ and $\mathbf{2^{12}}$ (e.g., 10 different truck sizes).
- Each worker is paid 100 TL per hour.

Note: For each task, if any empirical measurement is needed (e.g., estimating time by running a simulation code), you need to repeat it five times and take the **average of the five runs**.

Task 1: Which Algorithm? – 60 points

Model each truck as an array that contains the trees it carries, and each number in the array as the heights of the trees. Assuming that one worker is responsible for sorting the trees inside a truck, find the total duration of labor work for sorting the trees in each truck (10 trucks, each with different sizes as described above) by using Insertion Sort, Selection Sort, Bubble Sort, and Quick Sort. Which one is a good fit for this problem?

Submission Materials: *1- Your time estimations in a 10 x 4 table with its related plots (duration vs size). 2- Results discussion.*

Task 2: An Assumption! – 20 points

One of the companies realizes that Bilkent requires the trees to be delivered in sorted order and attempts to sort them before shipment. However, upon delivery, they discover a bug in their system, making the cargo of trucks **almost** sorted. An array is considered almost sorted if 6% of its items are far from their correct positions. Given this new condition, which sorting algorithm (the algorithms mentioned in the previous task) would you recommend for minimizing labor without any empirical results and why? Experiment with all algorithms mentioned in the previous task for the almost sorted trucks. Discuss the algorithms' results, and justify your recommendation based on them.

Submission Materials: *1- Your recommended algorithm and the reason behind it. 2- Your time estimations in a 10 x 4 table with its related plots (duration vs size). 3- Results discussion.*

Task 3: Increasing the number of workers – 20 points

Time plays a crucial role in planting the trees. However, according to weather forecasts, some days will be rainy, and it is not possible to plant trees on a rainy day. Consequently, the administration decided to double the number of workers so that two people could work on each truck. However, they have yet to determine if doing this will lead to any benefit or not.

You need to answer the concerns of the administration by suggesting a suitable and efficient algorithm that is capable of being leveraged by multiple (in this particular case, two) people. It is essential to calculate the cost of sorting each truck and compare it with the best algorithms' cost from Task 1. Additionally, you need to find the maximum and minimum time estimation for each worker without considering their idle time (one worker may do more work than the other worker).

Hint: *Think about a recursion tree, and try to model this problem based on the tree.*

Note: *You can answer this question by either hypothetical or empirical methods.*

Note: *It is allowed to use the results from previous tasks.*

Submission Materials: *1- Your suggested algorithm and the reason behind it. 2- Your maximum and minimum time estimations for each worker. If you are using a hypothetical method, it is possible to report a dependent function instead of an exact number. In the case of choosing the empirical method, you need to report a table that has the average time estimated to sort a truck and the cost of it. 3- Results discussion and reasoning based on them.*

Instructions:

- This homework is prepared by TA **Masoud Poorghaffar** m.poorghaffar@bilkent.edu.tr. Please direct your questions regarding this homework to him.
- The assignment is due **April 8, 2024, 23:59.** You should upload your homework to the upload link on Moodle before the deadline. No hardcopy submission is needed. The standard rules about late homework submissions apply. Please see the course web page for further discussion of the NO late homework policy as well as academic integrity.
- Any algorithm that you mention or discuss in your report must be implemented by you personally. This means that you are responsible for writing the code for these algorithms, and they should not be copied or sourced from external references or sources. You can reuse the codes from the slides.
- You must submit a report (as a PDF file) that contains all information requested above (plots, tables, discussion, etc.) and a cpp file that contains the functions that you used in your simulations.
- You should prepare your report using a word processor and convert it into a PDF file. In other words, do not submit images of handwritten answers. Handwritten answers and drawings will not be accepted. In addition, DO NOT submit your report in any other format than pdf.
- This is an individual assignment for each student. That is, you are **NOT** allowed to share your work with your classmates.

- You are required to write the materials mentioned in the "***Submission Materials***" section of each question in your report. Otherwise, you will be penalized.
- To submit the homework file, please compress your report (pdf file) and source code as a gzipped TAR file or a ZIP file with the name `CS201_HW2_Firstname_Lastname`. Please do not use Turkish letters in your file names.
- If you do not follow the submission guidelines and specifications (codes, report, etc.), it will lead to a significant grade deduction.
- Report your outputs in tables similar to the following table. (The number of columns may vary from question to question.)

n	Insertion Sort	Selection Sort	Bubble Sort	Quick Sort
2^3				
…				
2^{12}				

- To have clean plots, use the **log** of estimated values for plotting them.
- You can use the following code for generating almost sorted arrays.

```cpp
#include <cstdlib>
#include <algorithm>
#include <iostream>

int *generate_random_array(int size) {
    // TODO: you have to implement this
}

int *generate_almost_sorted_array(int size) {
    srand((unsigned int)time(NULL));

    int *random_array = generate_random_array(size);
    // TODO: here, you have to sort this array.

    int swap_count = size * 0.03;
    for (int i = 0; i < swap_count; i++) {
        int first_index = std::rand() % size;
        int second_index = size - first_index;

        int temp = random_array[first_index];
        random_array[first_index] = random_array[second_index];
        random_array[second_index] = temp;
    }

    return random_array;
}
```

Good Luck