CS-3364 Design and Analysis of Algorithms

Project 1

1 Goal

The goal of the project is to implement the selection sort, heapsort, and mergesort algorithms and verify that theoretical differences in efficiency correspond to reality.

2 Comparison

The steps to compare the algorithms are the following:

1. The algorithms must be compared using arrays of randomly chosen integers of sizes 50.000, 100.000, 150.000, and 200.000. Execute every algorithm at least three times on each array.[1] Close any other unnecesary program while running your algorithm (e.g., virus scanners, music players, etc.). Report these times and their mean in a table.

2. Plot mean execution times versus array size for each of the algorithms and analyze the resulting curve to identify if the shape is as expected (i.e., if the curves of the $\Theta(n^2)$ algorithms have a parabolic shape and the curves of the $\Theta(n)$ and $\Theta(n \log n)$ algorithms are almost straight).[2] (Do not forget to indicate the units used in the time axis, i.e., s, ms, etc.).

3. Plot the curves combined in a new set of axis. Show the time in a logarithmic scale to counteract the large execution time differences between the quadratic and (almost) linear algorithms. Analyze if the relation between the curves is as expected.

This process must be explained and analyzed in detail in the project report (see Section 3.1).

[1] If the largest execution time is at least 1.5 times the smallest, the operating system was probably doing something else while running your task. If this happens ignore that execution and run the algorithm again. Repeat as necessary until the largest execution time is no more than 1.5 times the smallest.

[2] Recall that $n \log n < n^{1+\epsilon}$ for $\epsilon > 0$ y n sufficiently large n.

3 Deliverables

3.1 Report

The following sections describe the expected content, languages, format and style of the report.

3.1.1 Contents

The parts of the reports must be those of a scientific article. They must contain (at least) the following sections: title, abstract, introduction, methodology, results, conclusions, and bibliography. However, the titles of the sections can change if necessary. Next, we describe the content of each section.

Title The title must be representative of content content of the paper. Avoid using interrogative titles like *What is the best sorting algorithm?*

Abstract An abstract summarizes the content of the paper in a few lines (a paragraph, to be exact). It summarizes the research questions, methodology used to answer it, results obtained, and main conclusions you arrived to (hopefully, the answer to the questions). Each section of the report is summarized here in one or two sentences to form a paragraph with four to eight sentences. The abstract is the first thing in the paper (after the title and authors' info), but is the last thing to be written (for obvious reasons).

Introduction The Introduction presents the research questions and a brief description of what will be done (the methodology) to answer them.[3] The predominant form of communication in the Introduction is text. Cites to bibliographical references are usually placed here.[4]

Methodology The Methodology presents the details of the experiments that permit to answer the formulated questions. In this section, a detailed description of the procedure should be the predominant component.

Results This section shows the results obtained from the experiments. Tables and figures are usually abundant here. [21 pts. See details in Section 2]

Conclusions This section states the conclusions that can be inferred from the results.[5] Text is the dominant form here.

Bibliography This section shows the information of the bibliographical sources cited in the text.[6]

[3] For this homework, the question should be whether the theory corresponds to practice (i.e., whether the asymptotically faster algorithms in theory, are so in practice).

[4] For this homework, a reference to the textbook is probably sufficient.

[5] Here you are expected to tell what are the best algorithms.

[6] For this homework, probably only the textbook.

3.1.2 Format

The reports must follow the format of the Institute of Electrical and Electronics Engineers for its Transactions journals (except the Transactions on Magnetics and Transactions on Dielectrics, which use other formats). The goal is that the student, besides learning to make technical and scientific reports, have the experience of limiting to a preestablished format (a common practice not only in academia but also in industry). Specifications of the format and LaTeX and MS-Word templates can be found in `https://www.ieee.org/publications_standards/publications/authors/author_templates.html`. LyX also provides a template with this format. (Go to File ▷ New from template and select `IEEEtran-Journal.lyx`. But be careful… some templates have similar names but different formats.)

3.2 Code

Implement the algorithms in the file *sort.c* published with this description. The functions headers must not be modified since the instructor will use a script to run them, and any change in the methods interface will make the script to fail. (However, you can add extra functions customized to your needs and call them in the body of the required methods.) The code must be compiled using the online compiler available at `https://www.programiz.com/c-programming/online-compiler/`. *If the code does not compile you will receive a grade of zero. If a particular algorithm does not correctly sort the arrays, you will receive a grade of zero for that algorithm.* To determine if a sorting algorithm is correct, we will run the following test:

```
for(int i=1; i<n; i++)
    if( A[i] < A[i-1] )
        cout << "Failed!";
```

4 Submission

This homework must be submitted on or before its due date through Blackboard. The student is responsible for checking the integrity of the submitted homework, specially if it is in a zipped file.[7] If you submitted multiple copies, only the last one will be considered. If the homework is submitted after the due date, the following police (specified in the syllabus) will be applied: *the grade received by a student who submitted a homework late will not be higher than the lower grade received by a student who submitted the homework on time.*

If you are having problems uploading your homework and the due date is approaching, send it by e-mail to the instructor (`arturoca@ttu.edu`). In case multiple submissions by email are received, only the first one will be considered.[8] The *late submission* policy mentioned in the previous paragraph applies to email submissions as well.

[7] Download the submitted file and unzip it to verify the integrity of its contents.

[8] The purpose is to discourage multiple email submissions.