Analysis of Algorithms 2020/2021

Practice 2

Your names,  Group.

|  |  |  |  |
| --- | --- | --- | --- |
| Code | Plots | Memory | Total |
|  |  |  |  |

**1. Introduction.**

Here you write a introduction and a previous discussion to the practice.

Practice number two has as objective try to determine experimentally the execution times of algorithms which use divide and conquer approach. These algorithms are MergeSort and QuickSort. The second one we are going to study it in a recursive form and in a non-recursive form to analyze the differences in time of execution, number of operations, etc. On each algorithm we will analyze tables of different sizes and we will compare the results obtained with the theoretical study of each algorithm.

In order to achieve these goals, we are going to use part of the code of practice 1. We will use files permutations.c, times.c, the makefile and, with some modifications, exercise4.c and exercise5.c.

**2. Objectives**

Here you indicate the work you are going to do in each section.

2.1 Section 1

Section 1 Objectives.

The objective of section 1 was to implement the function MergeSort in the file sorting.c. The prototype of the necessary functions will be int mergesort(int\* table, int ip, int iu) and int merge(int\* table, int ip, int iu, int imiddle). Both functions return ERR in case of error and the number of BO that have been made if the table have finished sorted.

We will use exercise4.c to check that the function MergeSort orders correctly.

2.2 Section 2

Section 2 Objectives.

In this section we will adapt exercise5.c using the MergeSort algorithm in order to obtain the average clock time and the average, minimum and maximum number of BO that have been executed depending on the size of the permutations. Also, we will compare these values obtained with the theoretical results of the algoritm.

2.3 Section 3

Section 3 Objectives.

2.4 Section 4

Section 4 Objectives.

2.5 Section 5

Section 5 Objectives.

**3 Tools and Methodology**

Here you describe the environment (Windows, Linux, MacOS) the tools you have used (Netbeans, Visual Studio, Eclipse, gcc, Valgrind, Gnuplot, Sort, uniq, etc) and the development methodologies and solutions to the probles that you have used in each section, so as the tests you have done to the developed programs.

3.1 Section 1

Methodology and adopted solutions for section 1

3.2 Section 2

Methodology and adopted solutions for section 2

3.3 Section 3

Methodology and adopted solutions for section 3

3.4 Section 4

Methodology and adopted solutions for section 4

3.5 Section 5

Methodology and adopted solutions for section 5

**4. Source code**

Here you write the source code **only the routines you have developed** in each section.

4.1 Section 1

4.3 Section 3

4.5 Section 5

**5. Results, Plots**

Here you write the results obtaind in each section, including the required plots.

5.1 Section 1

Results for section 1.

5.2 Section 2

Results for section 2.

Plot comparing the best, worst and average BOs for Mergesort, comments to the plot.

Plot with the average clock time for Mergesort, comments to the plot.

5.3 Section 3

Results for section 3.

5.4 Section 4

Results for section 4.

Plot comparing the best, worst and average BOs for Quicksort, comments to the plot.

Plot with the average clock time for Quicksort, comments to the plot.

5.5 Section 5

Results for section 5.

Plot comparing the average clock time for both versions of Quicksort routine, with and without tail recursion.

**5. Answers to theoretical Questions.**

Here you answer to the thoretical questions in the practice.

5.1 Question 1

5.2 Question 2

5.3 Question 3

5.4 Question 4

**6. Final Conclusions.**

Final Discussion about the practice and the obtained results.