

# Programming and Data Structures 2025

## Exercise 6

### Algorithm Efficiency

#### Overview

---

The purpose of this laboratory exercise is to learn about algorithms and their complexity in relation to data sizes and data quality, i.e., to evaluate the Big-Oh for the different algorithms and cases. You have been given a framework for testing algorithms as a zip-archive, your task is to implement algorithms and running the framework to test their complexities and report on your findings.

#### Assignment

---

Implement the five algorithms Bubble sort, Insertion sort, Quick sort, Linear search and Binary search, test them using the framework and write a short report on the results in relation to the theoretical complexities.

```
=====
a) Menu
b) Exit

c) Bubble sort best case
d) Bubble sort worst case
e) Bubble sort average case

f) Insertion sort best case
g) Insertion sort worst case
h) Insertion sort average case

i) Quick sort best case
j) Quick sort worst case
k) Quick sort average case

l) Linear search best case
m) Linear search worst case
n) Linear search average case

o) Binary search best case
p) Binary search worst case
q) Binary search average case
-----
```

#### Work flow

---

**Implement algorithms** – Implement the five algorithms in algorithm.c, abiding by the signatures defined in algorithm.h. Do not change any other file than algorithm.c.

**Evaluate complexity** – Run the program with all 15 options to find out the actual complexities of all combinations of algorithms and data. The program can be made using 'make' on the enclosed Makefile. To run the program after making, use './main'.

**Write report** – The report should be about 4-5 pages including figures and tables, and include the following headlines:

- Title and author
- Introduction

- Description of the five algorithms along with their theoretical complexities for best, worst and average case and how these cases arise for each algorithm.
- Results from running the program (experimental results), presented as tables and possibly graphs/diagrams.
- Discussion on the results from the experiment in comparison to the expected theoretical complexities.
- Summary and Conclusions
- Lessons learned

Introduction and description of algorithms should be about one page, Discussion, Summary and Conclusions plus Lessons learned should also be about one page. The results part might vary in page count depending on the size of tables and if figures or graphs are used, but 15 tables will take some space so this section will be the largest in the report. I think it is difficult to do this part properly in less than two pages with only tables, so a minimum total page count for the report is about 4 pages. You may **not** use AI-tools to write the report.

### Demonstration

---

You should hand in your algorithms.c-file along with the report, without any zipping or other compression, just the .c-file. No testrun is done in the lab rooms, we manually test your algorithms offline.

### Report

---

Hand in the report as a pdf-file.

### Grading

---

Points are given for:

- Correct implementation of algorithms (1p)
- Report
  - results 0.5p
  - discussion 0.5p
  - summary and conclusions + lessons learned 0.5p
  - overall greatness 0.5p