School of Computer Science
University of Guelph

# CIS*3490 The Analysis and Design of Algorithms

Winter 2021
Instructor: Fangju Wang

## Assignment 3 Guide

You can develop your programs using any C system, as long as your programs can be correctly compiled and executed on the Linux system in SoCS.

You are allowed to use standard library functions. Your programs should be submitted as a tar file containing files like

readme.txt, main.c, P11.c, P12.c, P21.c, P22.c, P23.c, makefile.

Any compilation error or warning will result in a mark deduction. There will be some marks allocated for documentation.

Each file should have a comment at the beginning containing your name, id, date, and the assignment name.

The *readme* file should contain the following:

- Name, id and assignment number
- A brief and clear description of how to compile and run each program.
- Comparison and analysis for Q2.4.

Each function should have a brief comment describing its purpose. Also, any section of code where it is not easily apparent what the code does should have a short comment.

C function `ftime()` can be used to get the system time usage.

Hints for individual questions:

**1** If you choose to read strings in `data_4.txt` as integers, you may use `sprintf()` to convert the integers into character strings.

**1.1** Examine every string read from `data_4.txt` to see if it is an anagram of the string that the user enters.

**1.2** You may use a double *presorting* method: sort digits of each string into its *signature*, and then sort the strings by signatures. The signature of a string is a string in which the characters/integers are sorted in ascending or descending order. The following are examples of signatures. The signature of string 1714636915 is 9766543111 or 1113456679, and the signature of 1365180540 is 8655431100 or 0011345568. Strings with the same signature are said to be in the same anagram set. Strings in the same anagram set stay together in the double-sorting result. For this assignment, you should use descending signatures. You may lose 0's if using ascending signatures.

**2.1, 2.2, 2.3** You can follow the algorithms in the textbook.