National Yang Ming Chiao Tung University
Department of Electrical Engineering
Data Structures, Fall 2021

December 15, 2021
Instructor: Mark P.-H. Lin
TA: Y.-H. Chen

Programming Assignment #6

Sorting

1 Problem Description

Sorting is one of the most frequent operations in computing. Many problems can be efficiently solved with sorted data. Among many sorting algorithms, the "quick sort" is one of the most efficient methods that can achieve the time complexity, $O(n \log n)$.

In this problem, you are asked to implement the "quick sort" algorithm. Given a list of n unsorted elements , the algorithm processes the list by divide and conquer while iteratively swapping two elements in the list until all elements are sorted. In this programming assignment, we assume that the pivot must be the key of the first element in each sub-sequence of the list during divide and conquer.

You are asked to count the number of the "interchange", or "swap", operations during the whole quicksort process in the lecture notes (Page 13), or in the textbook (Program 7.6). The interchange is NOT counted for the same element. An example is given below:

An example Quicksort Process						
Input data	9	1	0	5	4	
Function "QuickSort":	**9**	**1**	**0**	**5**	**4**	// left = 0; right = 4; pivot = 9;
Function "interchange":	**4**	**1**	**0**	**5**	**9**	// i = 4; j = 4; **Interchange 4 and 9**;
Function "QuickSort":	**4**	**1**	**0**	**5**	9	// left = 0; right = 3; pivot = 4;
Function "interchange":	**0**	**1**	**4**	**5**	9	// i = 3; j = 2; **Interchange 0 and 4**;
Function "QuickSort":	**0**	**1**	4	5	9	// left = 0; right = 1; pivot = 0;
Function "interchange":	**0**	**1**	4	5	9	// i = 1; j = 0; **NO interchange!!**
Function "QuickSort":	0	1	4	5	9	// left = 0; right = -1
Function "QuickSort":	0	1	4	5	9	// left = 1; right = 1
Function "QuickSort":	0	1	4	5	9	// left = 3; right = 3
Function "QuickSort":	0	1	4	5	9	// left = 4; right = 4
Total # of interchange (swap): 2						

2 Input and Output Formats

The input file gives a list of unsorted elements' keys, which are all integers, in different lines. You will need to output the number of swap operations in the output file after applying quick sort with the pivot which is always the first element in each subsequence.

sample1.in	sample1.out
9 1 0 5 4	2

sample2.in	sample2.out
23 17 19 13 31 37 29	3

3 Command-line Parameter

In order to correctly test your program, you are asked to add the following command-line parameters to your program.

[executable file name] [input file name] [output file name]

(e.g., StudentID.exe sample.in sample.out)

4 Submission Information

Your program must be written in the C/C++ language, and can be compiled on the Linux platform. The source files of your program must be named with "[your student ID].h" and "[your student ID].cpp". The executable file name of your program must be "[your student ID].exe". To submit your program, please archive both executable and source files of your program into a single zip file, named "[your student ID].zip", and upload to E3.

5 Due Date

The zip file must be submitted through E3 before 23:59, December 26, 2021.

6 Grading Policy

The programming assignment will be graded based on the following rules:

- Pass sample input with compilable source code (50%)

- Pass four hidden test cases, which $n \leq 1,000,000$ (40%)
- Pass the fifth hidden test cases, which $n \leq 10,000,000$ (10%)

The submitted source codes, which are copied from or copied by others, will NOT be graded. There will be 25% penalty per day for late submission.