CS207-ALGORITHMS LAB
Lab 3

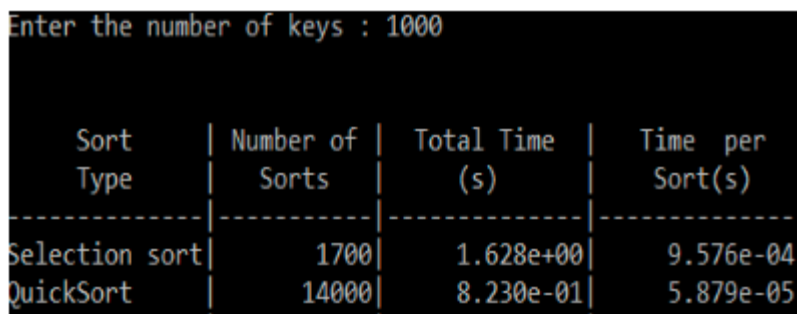---
NOTE:

• This lab is to be completed individually. Do not share your work or code with anyone else.

• You can use any programming language that you like; we suggest Python, C++, or C.

• For all of our labs, please avoid using Google to find suggestions or solutions. The goal is to use your own brain to work these problems out, which will help you develop the skills to do well in the exams and, more importantly, become a substantially better computer programmer.

• Save your work in your N-drive.

• You have to make lab report in hard copy.

Problem 1: Write Program to compare the execution times of linear search and binary search. Output of your code must look like the given image below.

A sample code given below to calculate execution time of a program:

```c
#include<stdio.h>
#include<time.h>
int main() {
        int i;
        double total_time;
        clock_t start, end;
        start = clock();
        //time count starts
        srand(time(NULL));
        for (i = 0; i < 25000; i++) {
                printf("random_number[%d]= %d\n", i + 1, rand());
        }
        end = clock();
        //time count stops

        total_time = ((double) (end - start)) / CLK_TCK;

        //calulate total time
```

```
        printf("\nTime taken to print 25000 random number is: %f",
total_time);

        return 0;

}
```

Problem 2: Write a program to sort a given set of elements using the Quick sort method and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator.

Problem 3: Write a recursive and non-recursive program to calculate Fibonacci numbers and analyze their time complexity.