



Department of Electrical and Software Engineering
Schulich School of Engineering

ENSF 694 - Principles of Software Development II
Summer 2023

Assignment#1

Due: July 09, Sunday, 11:59 PM

Submission: github link of the codes in the d2l dropbox 'Assignment#1'

The objective of this assignment is to apply your understanding of the searching and sorting algorithms.

Weighting: This assignment is out of 20 marks and is worth 10% of your overall grade.

Grading:

- All codes must be complete and compile without any errors.
- The codes should work for not only the given sample inputs but also any inputs of the same data types.

Inputs:

An input text file, and an integer search key.

The user will be prompted to enter the file name and search key. Inputs entered directly in the code will not be accepted and marked as 0 (zero).

Outputs:

List of all characters present in the text file with their corresponding frequencies (in any order),
Sorted array of frequencies of the characters present in the input string,
decision on search key FOUND/NOT FOUND, and
search key index if found.

Tasks:

The steps of the assignment are as follows-

1. Prompt and take the input file name from the user
2. Prompt and take the search key from the user
3. Read the input file content into a string
4. Compute the frequency of each character in the text
5. Show the characters and their frequencies (Do not include characters that are not present in the input text)
6. Create an integer array with the frequencies
7. Sort the array using "Merge Sort" (Any other sorting will not be accepted)
8. Show the sorted array
9. Apply "Binary Search" on the sorted array to search for the key (Any other searching will not be accepted)
10. Show the search results

Sample Input-Output:

Sample paragraph: Input1_A1.txt

Not so long ago, anyone who had heard the word algorithm was almost certainly a computer scientist or mathematician. With computers having become prevalent in our modern lives, however, the term is no longer esoteric. If you look around your home, you will find algorithms running in the most mundane places: your microwave oven, your washing machine, and, of course, your computer. You ask algorithms to make recommendations to you what music you might like or what route to take when driving. Our society, for better or for worse, asks algorithms to suggest sentences for convicted criminals. You even rely on algorithms to keep you alive, or at least not to kill you: the control systems in your car or in medical equipment. The word algorithm appears somewhere in the news seemingly every day.

Sample search key: 37

User prompt after running the code:

Enter input file name: Input1_A1.txt

Enter search key: 37

Output:

List of all characters in the text:

space - 133

, - 11

. - 7

: - 2

l - 1

N - 1

O - 1

T - 1

W - 1

Y - 2

a - 42

b - 2

c - 22

d - 14

e - 69

f - 6

g - 17

h - 27

i - 42

k - 8

l - 26

m - 31

n - 42

o - 72

p - 9

q - 1

r - 50
s - 37
t - 50
u - 26
v - 11
w - 14
y - 19

Sorted frequency array:

[1, 1, 1, 1, 1, 1, 2, 2, 2, 6, 7, 8, 9, 11, 11, 14, 14, 17, 19, 22, 26, 26, 27, 31, 37, 42, 42, 42, 50, 50, 69, 72, 133]

Search key '37' FOUND.

Search key '37' is at index 24.