Data Structure

Homework 3

Deadline: 2020/12/14 Mon. 23:55

Task 1: (I/O: 35 points, coding style: 5 points)

Given a text (≤ 1000 characters), please calculate each character's frequency (only 52 alphabets) and using Huffman coding. What is the number of bits after the final coding (the length of the ciphertext after coding)? Your program has to read till text equals `-1`. The program must be implemented **by linked list**, or you will get zero points.

Example:

Input	Output
`aaAb ccc`	`> 13`
`There's a car.`	`> 28`
`-1`	

Task 2: (I/O: 35 points, coding style: 5 points)

Given a number n (n = -1 or $1 \le n \le 2^{10}$) represents n integers. The next row is given n serial integers ($-2^{31} \le$ values $\le 2^{31} - 1$), representing the nodes. Please write a program to build a binary search tree and show the preorder and postorder traversals. The root is the first integer of the serial integers. If the upcoming integer equals to an existing node, put it on its right (right-child). For other cases, put the smaller ones on left (left-child), bigger ones on right (right-child). Your program has to read till n equals `-1`. The program must be implemented **by linked list**, or you will get zero points.

Example:

Input	Output
`8` `7 3 4 8 2 6 5 1`	`> Preorder: 7 3 2 1 4 6 5 8` `> Postorder: 1 2 5 6 4 3 8 7`
`3` `100 1 10`	`> Preorder: 100 1 10` `> Postorder: 10 1 100`
`-1`	

Put the files below in the folder (folder name: studentID), and compress this folder as **"studentID.zip"**.

1. Two source code files (filename: studentID_1.c, studentID_2.c)
2. One report with your coding environment (OS, IDE, ...), problems you encountered, and references. (filename: studentID.pdf) (10 points)

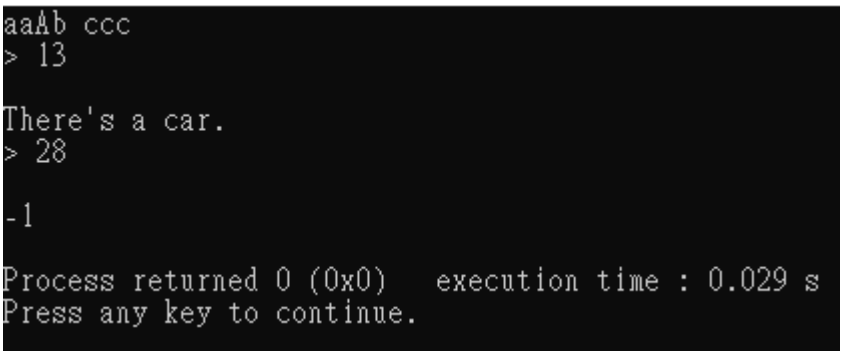
All the file names are correct, or you'll get zero points. (10 points)

You must hand in the assignment on time, or you will get zero points.

Warning: We encourage you to discuss assignments with each other. However, you have the responsibility to finish the assignments individually. **Do not copy others' assignment, or you will get zero points.**

Expected result:

(1)



(2)