Introduction to Software Design, Assignment Week_14-1

Deadline: **2023-06-10 23:59 (No score for late submission)**

- *Submit your homework through your zip file at the LMS assignment section. Here is an example.*

```
+ 12694_Assignment14-1_[STUDENT-ID].zip
  + 1.c
  + 2.c
  + 3.c
   ….
```

  - The zip file name format is like ***12694_Assignment[Assignment number]_[Student-ID].zip***
    - Student-ID example : 2024123456
  - The file format is like ***<filename>.c***
  - When uploading, delete the ***.vscode*** folder and the ***.exe*** file and upload it

You have to submit your solution to your zip file before the deadline.

1. Write a C program that takes two integers from a user, stores them to a `Point` struct variable, and prints `xpos` and `ypos` fields of the `Point` struct variable. You should dynamically allocate memory for the `Point` struct variable. Your program should work as the following example. (⏎ represents an 'Enter' keystroke from a user).

A.
```
typedef struct
{
    int xpos;
    int ypos;
} Point;
```

B.
```
(Example)
1 2⏎
1 2
```

C. Submission file: one C source file (File name is 1.c)

2. Write a C program that finds a specific word in the given words. Your program should work as the following example. (⏎ represents an 'Enter' keystroke from a user).

   A. Take a single English word from the user in each line until the given input is "end." The number of input words is a no limit and the length of each word cannot exceed 20.

   B. If the word already taken appears again, then show the warning message "This word is in the list."

   C. When the user inputs "end", shows the number of given words and their contents.

   D. Determine whether the given word is in the history list and print an appropriate message based on the search result. Repeat this process until the input is "end," as in A.

   E. Don't use realloc() and calloc() function.

   F.

```
(Example)
Enter a word (Enter 'end' to quit): tiger⏎
Enter a word (Enter 'end' to quit): dog⏎
Enter a word (Enter 'end' to quit): cat⏎
Enter a word (Enter 'end' to quit): cat⏎
This word already exists. Please enter another word.
Enter a word (Enter 'end' to quit): eagle⏎
Enter a word (Enter 'end' to quit): mouse⏎
Enter a word (Enter 'end' to quit): end⏎
5 words in the list:
tiger dog cat eagle mouse
Enter a word to search (Enter 'end' to quit): tiger⏎
This word is in the list.
Enter a word to search (Enter 'end' to quit): bird⏎
This word is NOT in the list.
Enter a word to search (Enter 'end' to quit): end⏎
```

   G. Submission file: one C source file (File name is 2.c)

3. Write a C program that takes students' name and scores as inputs and draw a star graph of students' scores. Your program takes name and score until the user give "END" as a name and 0 as a score. Students' names are up to 10 characters and the number of students has no limit. Your program should work as the following example. (⏎ represents an 'Enter' keystroke from a user).

   A. Print one '*' per 5 points and drop the remainder. (*ex.* "**" for 12 points and "*****" for 20 points)

   B. The array should be declared in `main()` and the graph should be drawn by calling `printScoreStars()` function.

   C. Prototype of `printScoreStars()` should be `void printScoreStars(Person* persons, int len)`.

   D.
```
typedef struct {
    char name[11];
    int score;
} Person;
```

   E.
```
(Example)
John 12⏎
Bob 20⏎
Amy 32⏎
END 0⏎
John   **
Bob    ****
Amy    ******
```

   F. Submission file: one C source file (File name is 3.c)