Creative Software Design, Assignment 2-2

Deadline: 2024-09-17 23:59 (No score for late submission)

- Submit your homework by uploading your zip file to the LMS assignment section. Below is an example.

```
13178_Assignment1-1_2024123456.zip
├─ 1.cc
├─ 2.cc
├─ 3.cc
└─ ...
```

- Your zip file name should follow this format:
  13178_Assignment[Assignment-number]_[Student-ID].zip
  - Ex. 13178_Assignment1-1_2024123456.zip
- Source files should be named as **<filename>.cc** *or* **<filename>.cpp**
- **You must submit your solution in the zip file before the deadline.**

1. Write a C++ program that replaces lowercase letters in a given string with the corresponding uppercase letters. (⏎ indicates that the user pressed the Enter key after entering the input)

   A. **Input**: A string that consists only of **lowercase letters**.

      - If the input string is "EXIT", the program should terminate normally.
      - If the input string contains one or more uppercase letters (except for "EXIT"), the program should print "warning" and ask for a new input string.

   B. Restriction

      - **Do not use the *toupper*() function**.

   C. Example output of your program (**Bold text** indicates user input):

```
Write lowercase string: hello⏎
HELLO
```

```
Write lowercase string: Hello⏎
"warning"
Write lowercase string: He!!~lo⏎
"warning"
Write lowercase string: hello⏎
HELLO
Write lowercase string: EXIT⏎
```

   D. Submission file: one C++ source file (File name: 1.**cc** or 1.**cpp**)

2. Write a C++ program to count the number of occurrences of specific patterns in a given input string. (⏎ indicates that the user pressed the Enter key after entering the input)

   A. **Input** : A single-line string. Run the program **two times** as an example.

      - The string consists of **only uppercase letters**. If the string contains any lowercase letters or invalid characters, the program should print "warning" and ask for another input string.

      Output : The first line shows the number of "IOI" patterns, and the second line shows the number of "OI" patterns.

   B. Example output of your program (**Bold text** indicates user input):

```
S = KOISIOI⏎
1
2

S = dfljeioioio⏎
"Warning"

S = IOIOI@!⏎
"Warning"

S = HFOIEIDJFKOIWJIOIOIDJKOIOI⏎
3
6
```

   C. Submission file: one C++ source file (File name: 2.**cc** or 2.**cpp**)

3. Write a C++ program that counts the number of **Sequence Words** in a given input. (⏎ means that the user pressed the enter key after entering)

A. **Definition of Sequence Word**

- A sequence word contains at least one character that is repeated consecutively. If there are no consecutive repeated characters, it is **not** a sequence word
- Examples of Sequence Words:

i. **caammmmee**: All characters (‘c’, ‘a’, ‘m’, ‘e’) appear consecutively, so it is a sequence word.

ii. **play**: Consecutive characters (‘p’, ‘l’, ‘a’, ‘y’) are not repeated, so it is a sequence word.

iii. **memory**: The character ‘m’ is not consecutively repeated, so it is **not** a sequence word

B. **Input** :

1. The first line contains an integer **N** ($1 \leq N \leq 100$), which is the number of words.
2. From the second line onward, each line contains one word.
   - The word should only consist of **lowercase** letters. If a word contains any uppercase letters or invalid characters, the program should print "warning" and prompt the user for a new word.
   - If **N ≤ 0** or **N > 100**, prompt for re-entry of **N**.

C. Example output of your program (**Bold text** indicates user input):

```
S = 3⏎
i⏎
like⏎
strawberry⏎
2
```

```
S = 4⏎
abcd⏎
kfjdkf⏎
k⏎
play⏎
3
```

```
S = 3⏎
you⏎
lIKe⏎
“warning”
love⏎
me⏎
3
```

D. Submission file: one C++ source file (File name: 3.**cc** or 3.**cpp**)

4. Implement the *func*() function to produce the following output based on the provided C++ code.

```
int main(){
    string str_1 = func();
    string str_2 = func("Hello", "Hanyang");
    string str_3 = func("Hello", "C++", "world")

    cout << str_1 << endl;
    cout << str_2 << endl;
    cout << str_3 << endl;
}
```

A. Example output of your program (**Bold text** indicates user input):

```
Hello world
Hello Hanyang
Hello C++ world
```

B. Submission file: one C++ source file (File name: 4.**cc** or 4.**cpp**)

5. Write down a C++ function multi(). It takes two int type parameters x and y, one of which has default value of 2.

A. Function multi

```
int multi(int _____, int
_______){
   // write your code.
}
```

B. Example

```
cout << mutli(3) << endl;
6
```

C. Submission file : one C++ source file (File name is 5.**cpp**)