EEE102 C++ Programming and Software Engineering II

Assessment 1

Fundamental of C++ language

Assessment Number	1
Contribution to Overall Marks	15%
Submission Deadline	Wednesday, 8-Mar.-2017, 23:55

How the work should be submitted?

SOFT COPY ONLY !

(MUST be submitted through ICE so that the TAs can run your programs during marking.)

Make sure your name and ID are printed on the cover page of your report.

Assessment Overview

This assessment aims at testing some basic concepts of C++ programming and initiates the routine of code development using the software development process (**SDP**), namely the five main steps of the software development process:

1. Problem statement: formulate the problem.
2. Analysis: determine the inputs, outputs, variables, etc
3. Design: define the list of steps (the algorithm) needed to solve the problem.
4. Implementation: the C code has to be submitted as a separate file. Just indicate here the name of the file.
5. Testing: explain how you have tested and verified your C program.

You will need to apply this methodology to each one of the following simple exercises.

What should be submitted?

A short ***report*** (up to a few pages of texts plus C source codes) detailing for all the questions of the assignment. The answer for each question should follow the SDP method:

a) SDP steps 1 to 3. (30% of the total marks for that question)

b) SDP step 4 (implementation): your C source code including the comments. (50%)

c) SDP step 5 (testing): you will explain how you have tested the correctness of your C program and will include some sample runs of your C Programs. (20%). **Testing result must be shown by screenshot.**

The report in Microsoft Word format **(.DOCX file)** and **C source code (with comments)**, for all questions should also be zipped into ***a single file***. (It is a good practice to include comments in your code stating the aim of the program, what are the inputs, what are the outputs, which algorithm is used, who is the author and so on.)

EXERCISE 1 (7 POINTS OUT OF 15)

Write a function

```
bool same_vec (vector<int> a, vector<int> b)
```

that checks whether two vectors have the same elements, ignoring the order and multiplicities. For example, the two vectors {1, 4, 9, 16, 9, 7, 4, 9, 11} and {11, 11, 7, 9, 16, 4, 1} would be considered identical.

Requirements:

1. Get the numbers from keyboard input;
2. The length of the vectors are unknown, it is only determined by the input.

EXERCISE 2 (8 POINTS OUT OF 15)

Write a function to search a character sequence pointed by a pointer (called "obj"), in another character sequence (called "source"). Return the pointer pointing to the found character. If there are more than one target found in source, return the pointer points to the first one.

Eg1: search for "C" in "ABCDEF", return the pointer point to 'C'.

Eg2: search for "Z" in "ABCDEF", return a NULL pointer.

Eg3: search for "CD" in "ABCDEF", return the pointer point to 'C'.

Eg4: search for "CF" in "ABCDEF", return a NULL pointer.

Eg5: search for "A" in "ABCAFC", return the pointer point to the first 'A'.

The function header is given by:

```
char *findC (char const *source, char const *obj);
```