Programming Fundamentals

Assignment 1 (10%)

[Deadline: 12:00:00 noon Fri 25th February 2022]

Question 1 (20 marks)

Write a C++ program to develop a simple calculator. The program prompts a menu (as shown in the figure below) of calculations for a user to choose (from 1 to 4). Then it asks the user to input two integers for calculation.

Assume the user's input is always correct and no data validation is required.

Below are the sample inputs and outputs:

Sample 1

```
MENU
        1. Divide. a/b
        2. Multiply. a*b
        3. Power. a^b
        4. Square root. sqrt(a)
Enter your choice:1
Enter two numbers:4 3
4/3=1.33333
```

Sample 2

```
MENU
        1. Divide. a/b
        2. Multiply. a*b
        3. Power. a^b
        4. Square root. sqrt(a)
Enter your choice:2
Enter two numbers:3 4
3*4=12
```

Sample 3

```
MENU
        1. Divide. a/b
        2. Multiply. a*b
        3. Power. a^b
        4. Square root. sqrt(a)
Enter your choice:3
Enter two numbers:4 2
4^2=16
```

Sample 4

```
MENU
        1. Divide. a/b
        2. Multiply. a*b
        3. Power. a^b
        4. Square root. sqrt(a)
Enter your choice:4
Enter a number:15
sqrt(15)=3.87298
```

Question 2 (20 marks)

Write a C++ program that asks the user to input a positive integer. Use it as a seed for generating a random number, x, where $x \in [2022, 24680]$. Display x and all its factors that are at least 2.

Assume the user's input is always a positive integer and no data validation is required. Your program should produce meaningful prompt and output messages.

Question 3 (25 marks)

Write a C++ program to repeatedly ask the user to input an integer between 1 and 99 and output the value in English. For example, if the input is 28, it should output "twenty-eight" (Note: there is a hyphen). Use only *if* and (nested) *if-else* structures in your implementation and you should realize the output English pattern to make your code concise. Here are the additional requirements of this question:

1. No character/string variables or any other means of storing character/string can be used.
2. The program will stop accepting more inputs if the input is -1.
3. If your code contains more than 35 *value comparison expressions*, only a maximum of 10 marks will be awarded when your program is correct. For example, "x == 5" is regarded as one *value comparison expression*. "x <= 10 || x >= 50" is regarded as two.

Assume the user's input is always correct and no data validation is required.

The input/output of the program should look like below:

```
6
six
28
twenty-eight
70
seventy
16
sixteen
-1
```

Question 4 (35 marks)

Write a C++ program that calculates the total number of days between two dates (inclusive), which are input by the user, in a format of `day month year`.

For example, if the input is `1 1 2022` and `11 4 2022`, the output is `101`, because January has 31 days, February has 28 days, March has 31 days and the remaining days in April is 11. So, it is 31 + 28 + 31 + 11 = 101. In your calculation, you need to consider the leap year, in which February has 29 days. Note that a leap year is divisible by 4, but not by 100, unless it is also divisible by 400. For example, the years 2000 and 2020 are leap years, but the years 2019 and 2100 are not.

Below are the sample inputs and outputs:

Sample Input	**Sample Output**
1 1 2022 11 4 2022	101
17 6 2018 4 2 2021	964
20 5 1999 9 9 2004	1940

Assume the user's input is always a valid date and no data validation is required.

Submission

Follow the steps below:

1. Create a folder and name it as A1_<student no>_<your name>.
   E.g., A1_12345678d_CHANTaiMan
2. Name the .cpp file as Q<question no>_<student no>_<your name>.cpp.
   E.g., Q1_12345678d_CHANTaiMan.cpp
3. Put the .cpp files into the folder.
4. Compress the folder (.zip, .7z, or .rar).
5. Submit the file to Blackboard.

In this assignment, only the C++ syntaxes and libraries covered in Lectures 1 - 4 and Labs 1 - 3 can be used.

Any wrong file naming and submission will be given ZERO mark in this assignment. If you are using Windows, the file extension may be hidden by the operating system. Follow the steps of below links to make sure the file extension is not hidden:

https://www.howtohaven.com/system/show-file-extensions-in-windows-explorer.shtml

If your program cannot be compiled, ZERO mark will be awarded for that program.

A maximum of 5 attempts for submission are allowed. Only the last attempt will be assessed.

The deadline of this assignment is **12:00:00 noon Fri 25th February 2022**. No late submission is allowed.

This assignment is an individual work. All work must be done on your own. Plagiarism is serious offence. The Moss (https://theory.stanford.edu/~aiken/moss/) system will be adopted for plagiarism checking. Submissions with high similarity, in terms of code patterns and structures, in addition to direct-copy-and-paste, will be treated as plagiarism. Copying code from web resources is prohibited as well. Any plagiarism cases (both copier and copiee) will be given ZERO mark in this assignment.