Assignment 1 [8 marks]

Lucky Number for Date and Digital Number Segmentation

Due: 15th Oct. 2023, 23:59 [Week 7]

1. This assignment contains one question with TWO parts. You are required to submit a complete C++ program (.cpp only) for each part on PASS before the due date.

2. You can submit as many times as you want before the deadline, and we will grade your latest version.

3. Only a small set of test cases is visible for the testing, and a more comprehensive set will be used for grading. In other words, passing all the visible test cases does not mean you can get full marks for this assignment. Try your best to test your program thoroughly. Hidden test cases will not be released.

4. The marking of each question is based on the percentage of the total test cases that your solution can pass. If your submitted solution leads to a compilation error on PASS, zero marks will be given to that question, and no manual checking of your solution is provided in such a case.

5. Late submissions will not be accepted. ALL submissions should be on PASS.

6. Plagiarism check will be performed.

7. You only need to use the material from Lectures 1 to 5. It is NOT necessary to include any other library, except <iostream>.

Part A. [4 mark] Lucky Number for Date

Write a program to calculate the "Lucky Number" based on a given date. Your program should perform the following tasks.

1. Prompt users for the given date.
   a. The input is in the format: yyyy mm dd.
      - Example: 2023 9 10
      - Example: 2023 12 25
   b. Verify if the birth date is valid.
      - Year should be from 1900
      - The combination of Year, Month, and Day should be valid
   c. If the date is invalid, print an error message and stop the program.
      - Example: If the date is 1899 1 1, display "Error!"
      - Example: If the date is 2023 2 29, display "Error!"
      - Example: If the date is 2023 13 1, display "Error!"
2. Prompt users for the given date.
   a. If the date is valid, print the message.
      - Example: If the date is 2023 1 1, display "The given day is 1st January 2023."
      - Example: If the date is 2010 7 21, display "The given day is 21st July 2010."
      - Example: If the date is 2021 3 2, display "The given day is 2nd March 2021."
      - Example: If the date is 2001 6 14, display "The given day is 14th June 2001."
3. Compute the lucky number for the given date.
   a. The lucky number is derived from adding all digits of the given date until it is a single digit.

      Example: If the given date is 2001 9 13, its lucky number is computed as (2+0+0+1+9+1+3) = 16 → (1+6)=7.
   b. Display the lucky number and stop the program.
      - Example: If the date is 2001 9 13, display “The lucky number is 7.”

Sample Input and Output [Be careful with the white space of in/outputs.]

Example 1

```
Please enter your birth date (yyyy mm dd): 1899 1 1

Error!
```

Example 2

```
Please enter your birth date (yyyy mm dd): 1899 2 29

Error!
```

Example 3

```
Please enter your birth date (yyyy mm dd): 2023 1 1

The given day is 1st January 2023.

The lucky number is 9.
```

Part B. [4 marks] Digital Number Segmentation

Write a program that reads a positive integer number (data type: int) as input and prints the number in segments of 3 digits. For each 3-digit segment, print the least significant digit (LSD) in English. Do not print the 3-digit segment if it is all 0.

1. You do not know the number of digits from the input in advance.
2. The input number will be no greater than INT_MAX, i.e., the maximum number that an int-type variable can represent.
3. We assume the input is valid, i.e., there is no need to check the correctness of the input.

Sample Input and Output

Example 1

```
Please enter an integer number: 2147883647

002 Two

147 Seven

883 Three

647 Seven
```

Example 2

```
Please enter an integer number: 1

001 One
```

Example 3

```
Please enter an integer number: 6000005

006 Six

005 Five
```

Example 4

```
Please enter an integer number: 0

No found!
```