Introduction to Engineering Computing
Spring 2020
Homework #2
Due: February 21st @ 5pm

Homework Description:

You are an analyst at the Central Intelligence Agency (CIA). You are assigned the job of developing a program that will analyze PIN numbers that are used to encrypt sensitive data. The CIA uses a very specific set of rules for PINs to identify non-CIA operatives who attempt to access the sensitive data (these non-CIA operatives will try PINs that do not conform to CIA rules). The rules are complicated enough that the CIA has assigned you the job of developing a program to test whether a PIN meets their rules. CIA personnel will then use the program to make sure the PINs they select to protect data conform to the rules.

Note: The program needs to check *first* whether the input is *valid* or not. If the input is a positive integer it is *valid*; if the input is a non-positive integer, it is *invalid*. You may assume that the user will input an integer value (no special characters/decimals).

If the input is *invalid*, you should print an error statement, e.g. "Invalid Input" to the user and then prompt the user to re-enter the PIN until a positive integer is entered, before checking any of the following PIN rules below.

For valid user inputs, the program will continue to check the following rules to determine if the PIN is acceptable.

The rules that your program must use to determine if the PIN is valid are as follows:

1. The PIN must be between 5 to 7 digits long (assume no leading zeros are entered – for example: 002345 is only 4 digits for our purposes).
2. The PIN cannot have the same digit in two consecutive positions (e.g. 12234 would not be a valid PIN)
3. The sum of the last 4 digits of the PIN must be a multiple of 3.
4. The value of ***n "choose" k***, where ***n*** is the largest digit and ***k*** is the left-most digit of the PIN, should be an even number. The expression ***n "choose" k*** is computed as

$$\frac{n!}{(n-k)!\,k!}$$

where the ***!*** denotes a factorial. For example, for PIN 645823, the largest digit n=8 and the left-most digit k=6.

$$\mathbf{8}\ \boldsymbol{choose}\ \mathbf{6} = \frac{8!}{(8-6)!\,6!} = \frac{8*7*6*5*4*3*2*1}{(2*1)*(6*5*4*3*2*1)} = 28.$$

Therefore 645823 would be a valid pin.

If the PIN follows the four rules above, you should report that the PIN is a CIA *compliant* PIN. If the PIN does **NOT** meet the rules, then your program should report that the PIN is *non-compliant* and print out at least one reason why it is not compliant according to the above rules. Your program must continually ask for a new PIN (no matter if the previous PIN was compliant or not) until a sentinel value is entered. The sentinel value will be negative one (-1). Once the sentinel value is entered, the program will end. You may **not** separately prompt the user for the individual digits of the PIN and you must use a **loop** to access the digits in the PIN.

Submit your *.cpp file to ICON by **5:00 PM** on **February 21st, 2020**.

HINTS:

- Develop your code step by step:
  - First, just write a code that continues prompting for (and reading in) a PIN until -1 is entered. Make sure that works before building on additional verifications.
  - Start with just one rule – pick the simplest one – and just test for that. Then add the second rule. And so on… Be sure you compile/run/test the code as you go.
- By hand, come up with several different test PIN values that should be compliant with all the rules. Make sure your code recognizes those as compliant. Similarly, determine some PIN values that comply with three of the four rules, but violate a fourth, so that you can make sure your code successfully identifies each of the four types of violations.
- For rule #4, when **n=k**, (***n choose k***) **= 1** because 0! = 1.
- In this homework, you can take advantage of integer division truncation. For example, to extract the digit in the tens position for number 213, you can divide 213 by 10 and apply the modulus or remainder operator (%) to the result: 213/10 = 21; 21%10 = 1.

GRADING Rubric (100 points total):

- 80 points for working code:
  - 10 points for continually asking for a PIN until sentinel value is entered
  - 10 points for validating the user input – i.e. non-negative integer
  - 50 points for correctly determining whether the PIN meets rules
    - 5 points for correctly determining whether the number of digits is between 5 and 7 (# of digits can equal 5, 6, or 7)
    - 10 points for accessing the PIN digits using a loop
    - 10 points for determining whether the same digit appears in consecutive positions
    - 10 points for correctly determining whether the sum of last 4 digits of the PIN is a multiple of 3
    - 15 points for correctly determining if ***n choose k*** is even
  - 10 points for reporting the compliance of the PIN and if the PIN failed, at least one reason why
- 20 points for style (See the Style guide under Content on ICON):
  - 5 points for indenting -- See Style Guide
  - 5 points for in-line comments -- See Style Guide
  - 5 points for comment blocks – See Style Guide
  - 5 points for meaningful variable names – See Style Guide

NOTES:

- If your program does not compile on a CSS Linux machine or you fail to submit a *.cpp file you will receive a zero on your homework. In addition, late homework will not be accepted.

DO NOT WORK TOGETHER! Students caught working together on this assignment will receive a **score of -100 points**!