Purpose

The purpose of this assignment is to give you practice with if statements and also to start using while loops.

Scenario

There are approximately 365.2422 days in a solar year, our calendar has 365 days in a standard year and occasionally has a leap year of 366 days to account for the fractional part. In -46 B.C the Julian calendar established that those years that are divisible by 4 will be leap years. This scheme over-estimates the length of the solar year by approximately one day every 128 years. By 1753 we had adopted the Gregorian calendar with a more complicated leap year formula that corrects for this.

Here is the definition of a Gregorian leap year:
For years 1753 A.D and after when Gregorian calendar is in effect, most years that are a multiple of 4 are leap years. However, if it is a multiple of 100, it is NOT a leap year unless it is ALSO a multiple of 400.

Problem

Write a program to read a year value from the input and determine whether it is a leap year or not. The process is the following: First prompt the user to enter a year and read in the value. Next, determine if the given year is a leap year or not and print out a suitable message which includes which calendar is in effect (Julian or Gregorian). You only need to check for years between -46 and 5000. Anything outside of this range is printed out as an Invalid year. Any value beyond values -9999 and 9999 will stop the program.

Your program continuously prompts the user for input (for Year) and stops only when the input is beyond -9999 or 9999. You must use a While Loop to achieve this.

The file you submit must be called leapyear.c – just to get you started there is an empty file by this name provided in the public folder.

Input

The input will come from standard input, that is, from a user at the keyboard. Input must be validated as per specification above. Keep accepting input until a number outside the specified range is entered.

Output

Output will be sent to standard output (the screen). Print the year entered by the user, which calendar is in effect and whether it is a leap year or not, and what calendar is in effect (or indicate it is invalid for years that are not valid as indicated above). (The value of the year is simply %d).

Output statement example: printf("Year %d: Gregorian Leap Year\n", year)

Examples:
Year 2400: Gregorian Leap Year
Year 100: Julian Leap Year
Year 5002: Invalid

Submission & Testing

On all your assignments, including this one, it is crucial that you test your program thoroughly. Do not add additional features that are not being asked for, since your program may not run against test inputs that I have created.

In Mimir we test several single values i.e these tests check for one acceptable year and terminate by providing a value outside the acceptable range. Your output for all these test cases should match mine exactly. You can view the expected outputs for the first few test cases in case you need to format it to match mine.

There are also test cases that test your loop by continuing to accept several years within the acceptable range to check your program flow. For these tests, the grade points may indicate zero (and you might see them as "passing") but there will be accompanying rubric items that will carry actual points when we grade them.

Grade Key

Name, comments, constants, variable names	8
Loops correctly until a number outside the range -9999 to 9999	23
Invalid years handled correctly	9
Gregorian calendar computation (years > 1752)	30
Julian calendar computation (years < 1753)	30