

# Blame Pope Gregory XIII

Andrew Rosen

## Abstract

This lab is an exercise that goes over if/else statements or switch statements, modular arithmetic, String indices, converting Strings to integers. It is adapted from Savitch's Java book.

## 1 Assignment

Write a program that, given a string as an input, tests if the given string is a valid date in the Gregorian Calendar.<sup>1</sup> Your program should output whether the given date is a valid. If the given date is not valid, report why.

- Dates in the US are formatted *MM/DD/YYYY*.
- Valid months are in the range [1, 12].
- September, April, June, and November each have 30 days.
- All other months but February have 31 days.
- February has 28 days, *except* on a leap year, where it has 29.
- Leap years are not as easy as you think.
  - A year not divisible by 4 is a normal year.
  - A year divisible by 4 is a leap year except ...
  - A year divisible by 100 is not a leap year except ...
  - A year divisible by 400 is a leap year.
  - For example:
    - \* 1644 is a leap year.
    - \* 1645 is not.
    - \* 1600 is a leap year.
    - \* 1700 is not.
  - You will need to think of a way to arrange the logic of these statements. Think about different ways to categorize leap years and not leap years.

---

<sup>1</sup>This is the most widely used civil calendar and it is the calendar used by the United States. It was commissioned and instituted by Pope Gregory XIII (1572-1585), although number of countries resisted using it well into the 1900s.

## 2 Hints

To complete this assignment, you will need to utilize some methods not used in class. Start by reading in the user's input using `input` method.

### 2.1 Slicing a String

Next we have to get the individual numbers out of the string. First to get a small portion of the string, we can use the slice notation, which allows us to save a small part of a string. It's of the format `someString[index:end]`

Here is an example:

```
s = "My name"
s2 = s[0:2] # holds the value "my"
s3 = s[3:6] # holds the value "nam"
s4 = s[3:]  # holds the value "name"
s5 = s[3]   # Not a slice, holds just "n"
space = s[2:3] # holds the value " "
```

Slicing a string returns a new string having the same characters as the portion of the original that starts a *index* start and up to, but not including *end*. Indices in Python start at 0, so the first character is at index 0, the second at index 1, and so on. If no *end* is specified, the substring goes until the end of the String.

### 2.2 Converting Strings to Integers

`int(someString)` converts a String into an `int`.

```
s = "123";
i = int(s); //stores 123
```

By combining this with some clever slicing, we can slice a string to get the parts of the String that represent the month, date, and year and convert each into an `int` using `int()` The rest of the program is logic!

### 2.3 Some Other Hints

The month is the first thing you want to check. The year only matters in February. Work on the other months first.

### 3 Grading Criteria

You don't need to use methods for this one, but you can if you want to.

**30 points** The program can tell the input is a date.

**30 points** The program correctly handles non-leap year dates.

**30 points** The program correctly handles leap year dates.

**10 points** The source code is reasonably formatted.

### 4 A Postscript About Time

This program may have been difficult. However, you have only scratched the surface. Time and dates are complicated by many factors, including daylight savings, odd time zones, conversion of historical dates from different calendars, governmental policy, and leap *seconds*, to name a few.

Take this lesson to heart. **Do not meddle with time.**

In your future career keep two things in mind when you must deal with time and dates. First, rely on libraries others have provided for you. Second, ask yourself: do you actually care about the actual time, or do you just care about the internal consistency?

This video from Tim Scott on Computerphile explains it thoroughly in the video linked in this sentence.