Joseph Remy,
Daniel Williamson

CS 126L: Section 3

27 November 2015

Lab 11: Date

1. Problem Statement

   Restated Problem: Develop a program that stores a date in the format MM-DD-YYYY and allows for adding "x" number of days or weeks. For extra credit, implement

   Assumptions:
   - Creating a Date class with str(), init(), and functions to add days OR weeks at a time
   - EC: Need to check the year at init and when the year is changed if it is a leap year or not.
   - Have a dictionary of the number of days in each month

   Requirements:
   - Must store the date in the constructor
   - Uses the date format MM-DD-YYYY
   - Able to add days or weeks at a time

2. Planning

   Explanation: Using a Date class with init, str, and methods for adding days or weeks at a time and checking the year for the leap year EC, we implemented the test code to confirm our expected results while also testing the extra credit section of our code.

   Steps:
   1. A Date object is created
   2. Date can now:
      a. print - Calls the str method of the object and prints out the date in MM-DD-YYYY format.
      b. addDays - Can add x days at a time to increment the date. Will properly add with the right number of days/months in a year, depending on whether or not it is a leap year.
      c. addWeeks - For x number of weeks, call addDays(7)
      d. checkYear - Calculations are used to determine whether or not the current year is a leap year

3. Testing and Implementation

   See attached "Lab11-Remy_Williamson-Date.py", "Date.py", and "Lab11-Remy_Williamson-Screenshot.PNG"

   Overall, the implementation of our code is pretty self explanatory. Otherwise, the code contains comments to explain each segment of the code.

   pep8:

```
C:\Windows\System32>pep8 "I:\CS126L\Lab 11\Date.py"

C:\Windows\System32>pep8 "I:\CS126L\Lab 11\Lab11-Remy_Williamson-Date.py"

C:\Windows\System32>
```

4. Reflection

   Overall, this lab is not that difficult, however, trying to change over to having the number of days in a month dictionary, *months*, from an instance variable to a class member variable was an interesting endevor. The reason for this change is to decrease unnecessary memory usage in the long run. For example, if we are creating a calendar of all the events happening at NAU and we are implementing our Date class, there would be 100s of Date objects with virtually the same instances of the *months* dictionary.

   Contact Information:

Joseph Remy Jr. - herobrine@nau.edu - 928-202-6229
Daniel Williamson - daw277@nau.edu - 928-830-8640