Andrew Sullivan
2/25/15
CMSI 402
Assignment 2

5.8

```
class DateToDay {
    public:
      char* convert( int dd, int mm, int yyyy ) {
        if( !(dd > 0 && dd <= lengthOfMonth(mm,yyyy) && mm > 0 && mm < 13) )
          throw;
      }
      int dayNumber = dd % 7;
      for( int y = 0; y < yyyy; y++ )
        dayNumber = (dayNumber + 365 + isLeapYear(y)) % 7;
      for( int m = 1; m < mm; m++ )
        dayNumber = (dayNumber + lengthOfMonth(m,yyyy)) % 7;
      switch( dayNumber ) {
        case 0:
          return( "Sunday" );
          break;
        case 1:
          return( "Monday" );
          break;
        case 2:
          return( "Tuesday" );
          break;
        case 3:
          return( "Wednesday" );
          break;
        case 4:
          return( "Thursday" );
          break;
        case 5:
          return( "Friday" );
          break;
        case 6:
          return( "Saturday" );
          break;
        default: throw;
      }
    }
    private:
      int lengthOfMonth( int mm, int yyyy ) {
        switch( mm ) {
          case 4: case 6: case 9: case 11:
```

```
            return( 30 );
          case 2:
            if( isLeapYear( yyyy ) )
              return( 29 );
            else
              return( 28 );
          default:
            return( 31 );
        }
      }

      isLeapYear( int yyyy ) {
        return( (yyyy % 4 == 0) && (yyyy != 0) );
      }
  };
```

5.9
You should tell the user that this user story can more than likely be broken down into smaller stories. If it is this length, it can most likely be several small user stories that can fit onto a 3x5 card.

6.6
One situation where a grep search could fail is when it returns too many queries to search through. If this happens, I would refine my search terms in an attempt to limit the amount of queries returned and hopefully, make them more accurate.

7.5
A propagating class is a class that serves as a middleman for other classes, relaying information between them. An example of this is mail carrier that relays information to and from other classes around it.

14.3
The purpose of the software plan is the first step in initial development. It summarizes the goals and circumstances of the project. It identifies important issues so that informed decisions can be made.

14.8
Product backlog for a small software that controls a washing machine.

ID	Feature	Does	Priority
1	On/Off	Turn the machine on or off	Required
2	Cycle Type	Pick what type of cycle (Ex: normal, fast, delicate)	High
3	Timer	Option to run for certain length of time	Moderate
4	Pause	A button to pause the cycle	Moderate
5	Cancel	Button to cancel a current cycle	High
6	Start	Button to start the selected cycle	Required

8.1
I would choose to incorporate my changes through polymorphism because it involves small changes, whereas creating a component class can involve large changes.

10.1
After completing 100% statement coverage, the software may still contain a bug. An example of this is with consecutive conditional statements. If we have three conditions in a row

```
Read (x); read (y);
if x > 0 then
        write (“1”);
else
        write(“2”);
if y > 0 then
        write(“1”);
else
        write(“2”);
```

We can test this with {<x=2, y=2}. This will cover statements write(“1”) for each of them but not the else statements.

10.2
Unit testing is the testing of a specific module of the software. It is used because this tests that a module correctly fulfills its responsibilities.

10.4
Regression testing is done after a change has been made, to retest the software. It finds whether the change inadvertently introduced new bugs into the intact parts of the software in an attempt to prevent them.

10.7
Unit testing tests specific modules of the software where functional testing tests the software as a whole.

10.12
The program's input is integers but it returns a double. We should either return doubles or change the return type to int.