

Stephen Smith
CMSI 402
Professor Johnson
2-25-15

Assignment #2

Problem 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 < yy; 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( (year % 4 == 0) && ( year % 100 != 0 || year % 400 == 0 ) );
    }
};
```

Problem 5.9

You should tell this person to divide the user story from the 10-page paper into smaller, more compact user stories that fit the 3x5 card. This is because anything that needs 10 pages to explain is most likely too intrinsic to contain in a single user story.

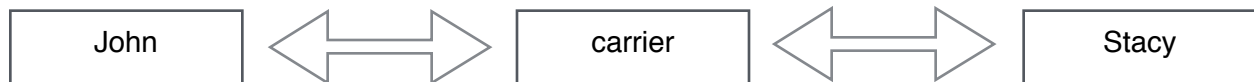
Problem 6.6

The grep tool can fail in many situations. A grep search can return no results. This would mean that the word being searched for does not appear in the code. Grep also fails when the sought after word appears in the code with a different meaning and does not indicate the correct concept location. Grep can even return so many queries that it is not feasible to go through all of them in an efficient manner.

In either case, I would follow the process of refining my search terms in such a way that would either widen or narrow the range of search terms that match the concept I am looking for.

Problem 7.5

A propagating class is a class that is the in-between of two or more other classes. It relays data from one class to another. The classic example is using a propagating class modeled after a mail carrier. The recipients can be their own individual classes, and all the carrier class does is relay messages from one person (or class) to another person (or class). This relationship can be seen below.



Problem 14.3

The purpose of the software plan is to summarize the goals and circumstances of the project. The software plan identifies the important issues so that informed decisions can be made. It allows managers to decide whether to move forward with a project or not.

Problem 14.8

ID	Feature	Description	Created By	Estimated Hours	Priority
95616	System On/Off	Implement a simple function for an on/off button that provides power to the washing machine	Jack Paulon	20	High
82459	Load Weight	Implement an option system that indicates the size of the laundry load on the machine (Heavy, normal, light)	George Washington	10	High
43623	Water Temperature	Implement an option system that indicates the temperature of the water (warm ,cool, cold)	Steve Jobs	10	High
90210	Begin Cycle Button	Implement a button whose function is to either start the washing machine or pause it to check the load (similar to a play button and a pause button)	Bill Gates	5	High
90045	LED Display	Implement a small LCD display to show how long it will take for the load to finish	Elon Musk	25	Medium
32354	Optional finished sound	Implement a button that sets whether there will be a loud sound to indicate the load is done	Scott Smith	5	Low
43617	Type of Cycle	Implement a dial system that indicates what type of cycle is desired (deep clean, heavy duty, whites, fast wash, normal, casual, colors, delicates, rinse and spin)	Sam Johnson	20	High
63069	Eco Cycle	Implement an EcoBoost button that will use less electricity and water	John Doe	10	Low
42557	Extra Rinse	Implement a button that will perform an extra rinse	John Doe	10	Low

Problem 8.1

I would choose (if it was correct for the situation) polymorphism because it leads to small change propagation. Creating a new component class often leads to larger change propagation that can extend to distantly connected classes.

Problem 10.1

No, the software could still contain bugs. The problem with statement coverage is that it does not identify bugs that arise from the control flow constructs. These could include things like compound conditionals or consecutive switch statements. This can be seen in the following code:

```
public int returnInput (int x, boolean c1, boolean c2, boolean c3) {  
    if (c1) {  
        x++;  
    }  
    if (c2) {  
        x--;  
    }  
    if (c3) {  
        x = x;  
    }  
  
    return x;  
}
```

The above code can be tested with the following:

```
int methodReturn = constructorInstance.returnInput(0, true, true, true);  
assertEquals(0, methodReturn);
```

Statement coverage does not cover the issue that arises when the first or second boolean parameter evaluates to false and the other true. In that case the returnInput function would not return the input it was given.

Problem 10.2

Unit testing verifies that the individual modules of a program are working to specification. One example would be testing a class. It is important that all of a class methods and properties work correctly. It is also important to test things like classes and their methods because they are the building blocks of a program. If one did not function correctly, then the entire program as a whole will most likely not function correctly.

Problem 10.4

Regression testing is the incorporation of tests from previously written components of a program to the tests for new components that might come along. It prevents previously written code from regressing in its desired functionality.

Problem 10.7

Unit testing is the testing of a specific module of the software. It makes sure that modules correctly fulfill their responsibilities. Functional testing verifies the functionality of the whole software package that is available to users.

Problem 10.12

While it might not be a bug, the language clearly supports curly braces for blocks of code so the if-else block should utilize curly braces (even if it is not required by the language). The language might also require the "return 0;" to be a "0.0" since the function is returning a double. The language might also interpret the "return x/y" as the whole number of times that y divides x. In other words, if x was always less than y, then the language might interpret that as 0.