

# COMP 410 - Spring 2015

## Programming Assignment 0



We seek to implement *sets of non-negative integers within a specified range*. The operations we will support are

- create an empty set object – we do so by specifying the upper bound  $U$  such that all elements to be stored in the set are in the range  $[0, \dots, U]$ ;
- insert a specified integer in the set object;
- set the set to equal the result of performing its union with another specified set object;
- set the set to equal the result of performing its intersection with another specified set object;
- determine whether a specified integer is within the set object; and
- print out the integers in the set object.

The *interface* is as follows:

```
//Sets of integers in the range [0 .. U], for some U that is specified in the constructor. (If unspecified, U defaults to 99)
public interface IntegerSet {
    static public final int range= 99;
    public void union(IntegerSet u1); //Union with the set u1
    public void intersection(IntegerSet u1); //Intersection with the set u1
    public void addToSet(int i); //if i < U, nothing is done
    public void removeFromSet(int i); //if not in the set, nothing is done
    public boolean elementOf(int i);
    public void printSet(); //print out the elements in the set
}
}
```

Design, code, and test a class

public class IntegerSetArrayImplementation implements IntegerSet{  
that implements the above interface by storing an integer set over the range  $[0, \dots, U]$  as a boolean array of length  $(U + 1)$ .