Anton Benfey & Paul Armstrong

CS1083 Lab 01

WishList.java source code:

```
import java.util.Scanner;


public class WishList {
	/**
	 The items on the customer's wish list, sorted by sku.
	*/
	private Item[] list;


	/**
	 Constructs a new WishList given a sorted array of Items.
	*/
	public WishList (Item[] listIn) {
		list = listIn;
	}


	/**
	 Constructs a new WishList by reading the number of items and then
	 the sorted list of item information using a Scanner; input format
	 consists of a line with the number of items, followed by a line for
	 each item containing values separated by commas
	*/
	public WishList (Scanner scin) {
		int count = scin.nextInt();
		scin.nextLine(); //read newline following the first int
		list = new Item[count];
		for(int i=0; i < count; i++){
			String s = scin.nextLine();
			Scanner scline = new Scanner(s);
			scline.useDelimiter(",");
```

```
                long sku = scline.nextLong();
                String name = scline.next();
                int priority = scline.nextInt();
                list[i] = new Item(name, sku, priority);
        }
}

/**
 Returns the number of items that appear in only one of the two
 wish lists (this one and the other one that is passed in as a
 parameter).
*/
public int findUnique (WishList other){
        //TO DO: Complete this method
        int counter = 0;
        boolean unique = true;

        for (int i = 0; i < list.length; i++){
                for (int j = 0; j < other.list.length; j++){
                        if (list[i].getSKU() == other.list[j].getSKU()){
                                unique = false;
                                break;
                        }
                }
                if(unique){
                        counter++;
                }

                if(!unique){
                        unique = true;
                }
        }
```

```
        counter += (other.list.length - (list.length - counter));


        return (counter);
}


/**
 Merges this wish list with another one (passed in as a parameter),
 producing a new sorted wish list.
*/
public WishList merge (WishList other){
        //TO DO: Complete this method
        Item[] newList = new Item[list.length + other.list.length];
        int counter = 0;
        int counter2 = 0;


        for(int i = 0; i < newList.length; i++){
                if((counter < list.length) && (counter2 < other.list.length)){
                        if(list[counter].getSKU() < other.list[counter2].getSKU()){
                                newList[i] = list[counter];
                                counter++;
                        }
                        else{
                                newList[i] = other.list[counter2];
                                counter2++;
                        }
                } else if((counter < list.length) && (counter2 >= other.list.length)){
                        newList[i] = list[counter];
                        counter++;
                } else if((counter2 < other.list.length) && (counter >= list.length)){
                        newList[i] = other.list[counter2];
                        counter2++;
                }
        }
```

```
            WishList finalList = new WishList(newList);
            return finalList;
        }

        public String toString(){
            String s = "";
            for(int i=0; i < list.length; i++){
                s += list[i].getSKU() + "\t" + list[i].getName() + "\t"
                    + list[i].getPriority() + "\n";
            }
            return s;
        }

}
```

Outputs: Output 1 was tested using the sample input from the assignment. We used this test case to see that our program outputted the same information, which it did.

```
11798411010,KitchenAid Stand Mixer,211039926010 Digital Kitchen Scale   3
11798411010     KitchenAid Stand Mixer  1
24179114710     Autumn Plaid Tablecloth 2
96796133410     Tan Cotton Blanket      2


11781701910     Cast Iron Round Griddle 2
11798009510     Espresso Machine        1
11798112010     NutriBullet Blender     1
11798411010     KitchenAid Stand Mixer  2

There are 6 items found in one wish list but not the other
Merged wish lists:
11039926010     Digital Kitchen Scale   3
11781701910     Cast Iron Round Griddle 2
11798009510     Espresso Machine        1
11798112010     NutriBullet Blender     1
11798411010     KitchenAid Stand Mixer  2
11798411010     KitchenAid Stand Mixer  1
24179114710     Autumn Plaid Tablecloth 2
96796133410     Tan Cotton Blanket      2
```

Output 2 is testing a scenario where every item is unique. This is important to test that the program can

Function with only unique items:

```
2
101,Cell,15
105,Book,20
101     Cell    15
105     Book    20

2
109,Shoe,255
156,Guitar,905
109     Shoe    255
156     Guitar  905

There are 4 items found in one wish list but not the other
Merged wish lists:
101     Cell    15
105     Book    20
109     Shoe    255
156     Guitar  905
```

Output 3 is testing a scenario where one list is of length 1, and the other list is greater than 1. This is important to see if the program functions with asymmetrical lists:

```
1
9,Clock,3
9       Clock   3

5
45,A,4
46,B,5
47,C,6
48,D,7
49,E,8
45      A       4
46      B       5
47      C       6
48      D       7
49      E       8

There are 6 items found in one wish list but not the other
Merged wish lists:
9       Clock   3
45      A       4
46      B       5
47      C       6
48      D       7
49      E       8
```

Output 4 is testing a scenario where both lists are empty. Theoretically, this should tell us that there are no unique items. It should not fail:

```
0

0

There are 0 items found in one wish list but not the other
Merged wish lists:
```

Output 5 is testing a scenario where there are NO unique items. This is important to test to assure the program does not rely on a unique item to function.

```
2
1,a,1
1,a,1
1       a       1
1       a       1

2
1,a,1
1,a,2
1       a       1
1       a       2

There are 0 items found in one wish list but not the other
Merged wish lists:
1       a       1
1       a       2
1       a       1
1       a       1
```