```
1 /** This class is a test class for testing the implementations: MyBad and MySet
2  * This tests the methods for the cardinality, isEmpty
3  * , isEmpty, union and intersection
4  *
5  *  *Name: Long Nguyen: Student # 5427059
6  *
7  * @version  1.0 (Mar. 2014)                                    */
8
9 package TestLists;
10 import java.util.Iterator;
11
12 import MULTISET.*;
13
14 public class TestHarness {
15
16   //The default constructor where it tests the methods
17   public TestHarness ( ) {
18
19      System.out.println("MySet :");
20      MultiSet <KeyedChar> Set = new MySet <KeyedChar>();
21      testList(Set);
22      System.out.println("");
23      System.out.println("");
24      System.out.println("MyBag :");
25      MultiSet <KeyedChar> Bag = new MyBag <KeyedChar>(100);
26      testList(Bag);
27      System.out.println("");
28      System.out.println("");
29      System.out.println("Checking the Union of Bag and Set ");
30      union(Bag,Set);
31      System.out.println("");
32      System.out.println("");
33      System.out.println("Checking the Intersection of Bag and Set  ");
34      intersection(Bag,Set);
```

```
35
36
37   }
38
39    /*Adding Elements to MyBag and MySet and checking if it's equal, finding
     the
40     * multiplicity and the cardinality of the set*/
41
42    private void testList ( MultiSet<KeyedChar> l ) {
43
44       System.out.print("Is the collection empty : ");
45       System.out.println(l.isEmpty());
46
47       System.out.println("Adding Elements to the Bag or Set ");
48
49          for ( char c='H' ; c>='A' ; c-- ) {
50             l.add(new KeyedChar(c));
51          };
52
53          System.out.print("The cardinality of the set: ");
54           System.out.println(l.cardinality());
55
56           System.out.print("The multiplicity of the set: ");
57           System.out.println(l.multiplicity(new KeyedChar('A')));
58
59           System.out.print("Is it equal : ");
60           System.out.println(l.equals(l));
61
62           System.out.print("Printing out the values of the Collect : ");
63            Iterator itr = l.iterator();
64              while(itr.hasNext()) {
65                 Object element = ((Keyed) itr.next()).getKey();
66                       System.out.print(element + " ");
67               }
```

```
68
69      }; // testList
70
71      /*This Method test the union of the two sets*/
72
73      private void union ( MultiSet<KeyedChar> Bag, MultiSet<KeyedChar>
    Set ) {
74
75          System.out.print("Adding elements to Bag : ");
76           for ( char c='A' ; c<='H' ; c++ ) {
77              Bag.add(new KeyedChar(c));
78              System.out.print(c + " ");
79            };
80            System.out.println("");
81
82            System.out.print("Adding elements to Set : ");
83            for ( char c='A' ; c<='L' ; c++ ) {
84              Set.add(new KeyedChar(c));
85              System.out.print(c + " ");
86            };
87            System.out.println("");
88
89            MultiSet <KeyedChar> myUnionBag;
90           myUnionBag =  Bag.union(Set);
91
92           System.out.print("Is Bag and Set equal : ");
93            System.out.println(Bag.equals(Set));
94
95            for ( char c='A' ; c<='L' ; c++ ) {
96              Set.add(new KeyedChar(c));
97            };
98
99           MultiSet <KeyedChar> myUnionSet;
100          myUnionSet =  Set.union(Bag);
```

```
101
102         System.out.print("The union of the MyBag and MySet : ");
103
104          Iterator bag = myUnionBag.iterator();
105            while(bag.hasNext()) {
106               Object element = ((Keyed) bag.next()).getKey();
107                    System.out.print(element + " ");
108             }
109
110            System.out.println("  ");
111
112            System.out.print("The union of the MySet and MyBag : ");
113             Iterator set = myUnionSet.iterator();
114               while(set.hasNext()) {
115                  Object element = ((Keyed) set.next()).getKey();
116                       System.out.print(element + " ");
117                }
118
119      }
120
121      /*This Method test the intersection of the two sets*/
122
123      private void intersection ( MultiSet<KeyedChar> Bag,
    MultiSet<KeyedChar> Set ) {
124
125         //adding elements to the bag/set
126         System.out.print("Adding elements to Bag : ");
127          for ( char c='A' ; c<='H' ; c++ ) {
128              Bag.add(new KeyedChar(c));
129              System.out.print(c + " ");
130            };
131            System.out.println("");
132
133          //adding elements to the bag/set
```

```
134          System.out.print("Adding elements to Set : ");
135          for ( char c='A' ; c<='L' ; c++ ) {
136             Set.add(new KeyedChar(c));
137             System.out.print(c + " ");
138          };
139          System.out.println("");
140
141          //Creating a MultiSet class and getting the intersection
142          MultiSet <KeyedChar> myIntersectionBag;
143          myIntersectionBag =  Bag.intersection(Set);
144
145          System.out.print("Is Bag and Set equal : ");
146           System.out.println(Bag.equals(Set));
147
148          for ( char c='A' ; c<='L' ; c++ ) {
149             Set.add(new KeyedChar(c));
150          };
151
152         //Creating a MultiSet class and getting the intersection
153          MultiSet <KeyedChar> myIntersectionSet;
154          myIntersectionSet =  Set.intersection(Bag);
155
156          System.out.print("The Intersection of the MyBag and MySet : ");
157
158          //iteratoring through the Bag and printing out the key values
159           Iterator myBag = myIntersectionBag.iterator();
160             while(myBag.hasNext()) {
161                Object element = ((Keyed) myBag.next()).getKey();
162                     System.out.print(element + " ");
163              }
164
165             System.out.println("  ");
166
167             System.out.print("The Intersection of the MySet and MyBag : ");
```

```
168
169             //iteratoring through the Bag and printing out the key values
170               Iterator mySet = myIntersectionBag.iterator();
171                 while(mySet.hasNext()) {
172                   Object element = ((Keyed) mySet.next()).getKey();
173                         System.out.print(element + " ");
174                 }
175
176       }
177
178
179    public static void main(String[] args) {
180
181       new TestHarness(); // new instance of Testharness
182
183    }
184
185 }
186
```