```
1  /**************************************************************
2  * Dalton Nofs                                                 *
3  * Login ID: nofs5491                                          *
4  * CS-102, Summer 2017                                         *
5  * Programming Assignment 3                                    *
6  * ListInterface class:  general front end for term              *
7  **************************************************************/
8  public interface TermInterface
9  {
10     /*************************************************************
11     * Method: isEmpty()                                           *
12     * Purpose: check to see if linkedList is empty                *
13     *                                                              *
14     * Parameters:                N/A                             *
15     * Returns: boolean:          if list is empty                 *
16     *************************************************************/
17     public boolean isEmpty();
18
19     /*************************************************************
20     * Method: size()                                              *
21     * Purpose: determine the size of linked list                  *
22     *                                                              *
23     * Parameters:                N/A                             *
24     * Returns: int:               the size of the array           *
25     *************************************************************/
26     public int size();
27
28     /*************************************************************
29     * Method: get()                                                *
30     * Purpose: get object from linked list at index               *
31     *                                                              *
32     * Parameters: int:           index                           *
33     * Returns: Object:           Object stored in index           *
34     *************************************************************/
35     public Course get(int index);
36
37     /*************************************************************
38     * Method: add()                                                *
39     * Purpose: add a object at specified index                    *
40     *                                                              *
41     * Parameters:                                                  *
42     *               int:          index                          *
43     *               Object:        Object to be placed            *
44     *                                                             *
45     * Returns: void:                N/A                           *
46     *************************************************************/
47     public void add(int index, Course item);
48
49     /*************************************************************
50     * Method: remove()                                              *
51     * Purpose: remove index postion and return object removed     *
52     *                                                               *
53     * Notes: calls func that can throw indexoutboundsexception     *
54     *                                                              *
55     * Parameters: int:           index                           *
56     * Returns: Object:           Object removed                   *
57     *************************************************************/
58     public Course remove (int index);
59
60     /*************************************************************
61     * Method: removeAll                                             *
62     *                                                              *
63     * Purpose: removes all nodes from array                        *
64     *                                                              *
65     * Parameters:             N/A                                 *
66     * Returns: void:          N/A                                 *
67     *************************************************************/
```

```
68      public void removeAll();
69
70      /*************************************************************
71      * Method: append                                             *
72      *                                                             *
73      * Purpose: appends the course to the end of the list         *
74      *                                                             *
75      * Parameters: Course: courseIn                               *
76      * Returns: void:            N/A                               *
77      *************************************************************/
78      public void append(Course courseIn);
79
80      /*************************************************************
81      * Method: removeAll                                           *
82      *                                                            *
83      * Purpose: removes all nodes from array                       *
84      *                                                            *
85      * Parameters:              N/A                               *
86      * Returns: String           the term                          *
87      *************************************************************/
88      public String getTerm();
89  }
```