

```

1  import java.util.*; // to use Scanner
2
3  // implementing the dynamic List ADT using array
4  // operations to be implemented: read, Modify, delete, isEmpty, insert, countElements
5
6  class List{
7
8      private int array[];
9      private int maxSize;
10     private int endOfArray;
11
12
13     public List(int size){
14         maxSize = size;
15         array = new int[maxSize];
16         endOfArray = -1;
17     }
18
19     public boolean isEmpty(){
20
21         if (endOfArray == -1)
22             return true;
23
24         return false;
25     }
26
27     public void resize(int s){
28
29         int tempArray[] = array;
30
31         array = new int[s];
32
33         for (int index = 0; index < Math.min(s, endOfArray+1); index++){
34             array[index] = tempArray[index];
35         }
36
37         maxSize = s;
38     }
39
40
41
42     public void insert(int data){
43
44         if (endOfArray == maxSize-1)
45             resize(2*maxSize);
46
47         array[++endOfArray] = data;
48     }
49
50
51     public void insertAtIndex(int insertIndex, int data){
52
53         // if the user enters an invalid insertIndex, the element is
54         // appended to the array, after the current last element
55         if (insertIndex > endOfArray+1)
56             insertIndex = endOfArray+1;
57
58         if (endOfArray == maxSize-1)
59             resize(2*maxSize);
60
61         for (int index = endOfArray; index >= insertIndex; index--){
62             array[index+1] = array[index];
63
64             array[insertIndex] = data;

```

```

65         endOfArray++;
66
67     }
68
69
70     public int read(int index){
71         return array[index];
72     }
73
74     public void modifyElement(int index, int data){
75         array[index] = data;
76     }
77
78
79     public void deleteElement(int deleteIndex){
80
81         // shift elements one cell to the left starting from deleteIndex+1 to
82         // endOfArray-1
83         // i.e., move element at deleteIndex + 1 to deleteIndex and so on
84         for (int index = deleteIndex; index < endOfArray; index++)
85             array[index] = array[index+1];
86
87         endOfArray--;
88
89     }
90
91     public int countList(){
92
93         int count = 0;
94         for (int index = 0; index <= endOfArray; index++)
95             count++;
96
97         return count;
98     }
99
100
101     public void print(){
102
103         for (int index = 0; index <= endOfArray; index++)
104             System.out.print(array[index]+" ");
105
106         System.out.println();
107
108     }
109
110 }
111
112 class DynamicListArray{
113
114     public static void main(String[] args){
115
116         int listSize;
117
118         Scanner input = new Scanner(System.in);
119
120         System.out.print("Enter list size: ");
121         listSize = input.nextInt();
122
123         List integerList = new List(1); // we will set the maxSize to 1 and double it as
124         // and when needed
125         for (int i = 0; i < listSize; i++){
126

```

```

127     int value;
128
129     System.out.print("Enter element # " + i + " : ");
130     value = input.nextInt();
131
132     integerList.insertAtIndex(i, value);
133 }
134
135 System.out.print("Contents of the list: ");
136 integerList.print();
137
138
139 // to read an element at a particular index (before delete)
140
141 int readIndex;
142 System.out.print("Enter an index to read (before delete): ");
143 readIndex = input.nextInt();
144 System.out.println("Value at " + readIndex + " is: " + integerList.read(readIndex) );
145
146 // to delete an element at a particular index
147
148 int deleteIndex;
149 System.out.print("Enter an index to delete: ");
150 deleteIndex = input.nextInt();
151 integerList.deleteElement(deleteIndex);
152
153 System.out.print("Contents of the list: ");
154 integerList.print();
155
156
157 // to read an element at a particular index (after delete)
158
159 System.out.print("Enter an index to read (after delete): ");
160 readIndex = input.nextInt();
161 System.out.println("Value at " + readIndex + " is: " + integerList.read(readIndex)
);
162
163 System.out.println("Number of elements in the list (before insert) is: " +
integerList.countList() );
164
165
166 // to insert an element at a particular index
167 int insertIndex, insertValue;
168 System.out.print("Enter an index to insert: ");
169 insertIndex = input.nextInt();
170 System.out.print("Enter a value to insert: ");
171 insertValue = input.nextInt();
172 integerList.insertAtIndex(insertIndex, insertValue);
173
174 System.out.print("Contents of the list: ");
175 integerList.print();
176
177
178 // to read an element at a particular index (after insert)
179
180 System.out.print("Enter an index to read (after insert): ");
181 readIndex = input.nextInt();
182 System.out.println("Value at " + readIndex + " is: " + integerList.read(readIndex)
);
183
184 System.out.println("Number of elements in the list (after insert) is: " +
integerList.countList() );
185
186

```

```
187 // to insert at the end of the list
188 System.out.print("Enter the element you want to insert at the end of the list: ");
189 insertValue = input.nextInt();
190 integerList.insert(insertValue);
191
192 System.out.print("Contents of the list: ");
193 integerList.print();
194
195
196 }
197
198 }
```

```
Enter list size: 5
Enter element # 0 : 99
Enter element # 1 : 88
Enter element # 2 : 22
Enter element # 3 : 11
Enter element # 4 : 33
99 88 22 11 33
Enter an index to read (before delete): 3
Value at 3 is: 11
Enter an index to delete: 3
Contents of the List: 99 88 22 33
Enter an index to read (after delete): 0
Value at 0 is: 99
Number of elements in the list (before insert) is: 4
Enter an index to insert: 0
Enter a value to insert: 11
Contents of the List: 11 99 88 22 33
Enter an index to read (after insert): 2
Value at 2 is: 88
Number of elements in the list (after insert) is: 5
Enter the element you want to insert at the end of the list: 44
Contents of the List: 11 99 88 22 33 44
```