CS218 Data Structures

Spring 2020 FAST-NU, Lahore

Instructor: Hafiz Muhammad Hamza

Assignment 2 – Linked Lists

Develop a template class **`ListArray`** which is an array based on Linked List. This array can hold data of any type. We'll work with the object of this class same as we work with arrays, but at the back end, the data is stored in a Linked List.

Provide the following functionalities:

Insert

Provide following functions for insertion:

`void insert(int val)` – inserts a new element at the end of the array.

`void insert(int list[], int n)` - inserts n number of elements at the end of the array.

`void insertAtFront(int val)`- inserts a new element at the front of the array.

`bool insertAt(int pos, int val)` – inserts a new element at the specified position. This element must be stored at the index specified by the user, and all the elements after that index must be shifted.

`array[pos] = val` - overload the subscript operator to store a value at the specified index. The value already stored at that index is overwritten.

Delete

Provide following functions for deletion:

`bool delete(int val)` – deletes the first occurrence of the value.

`bool deleteAt(int pos)` – deletes the value at the specified index.

`bool deleteFirst()` – deletes the first element of the array.

`bool deleteLast()` – deletes the last element of the array.

`bool deleteRange(int begin, int end)` – deletes all elements of the indices in the range begin to end, inclusive.

The return type of all the deletion functions is **`bool`** which will tell whether the deletion was successful or not.

Search

Provide a function `int search(int key)` which returns the index of the first occurrence of the key. If the key is not found, it returns -1.

Print

Provide a function `void print()` to print the whole array.

Reverse

Provide a function `void reverse()` to reverse the order of the elements in the array.

A sample run of this program is given below:

```
ListArray<int> myArray;

myArray.insert(4);
myArray.insert(11);
myArray.insertAtFront(7);
myArray.print();

myArray.insertAt(1, 9);
myArray.insertAt(1, 1);
myArray.print();

myArray[0]=2;
myArray.print();

myArray.delete(9);
myArray.delete(6);  //returns false
myArray.deleteAt(2);
myArray.print();

myArray.deleteFirst();
myArray.deleteLast();
myArray.print();

int list[10]={2,3,5,7,11,13,17,19,23,29};
myArray.insert(list, 10);
myArray.print();

myArray.deleteRange(5,8);
myArray.print();

myArray.reverse();
myArray.print();

//Output
```

```
7 4 11
7 1 9 4 11
2 1 9 4 11
2 1 11
1
1 2 3 5 7 11 13 17 19 23 29
1 2 3 5 7 23 29
29 23 7 5 3 2 1
```