

Traverse and search

I am hoping that you can see that **you can only traverse this single linked list one way**. This is slow unlike arrays where you can use the indices to go straight to a data element

Operations on any but the first element requires traversal of the list. As you do this you have to make sure you don't crash the program if the number you are looking for is not in the list.

```
//Java code for Linked List implementation
```

```
import java.util.*;
```

```
public class Test
```

```
{
```

```
    public static void main(String args[])
```

```
    {
```

```
        // Creating object of class linked list
```

```
        LinkedList<String> object = new LinkedList<String>();
```

```
        // Adding elements to the linked list
```

```
        object.add("A");
```

```
        object.add("B");
```

```
        object.addLast("C");
```

```
        object.addFirst("D");
```

```
        object.add(2, "E");
```

```
        object.add("F");
```

```
        object.add("G");
```

```
        System.out.println("Linked list : " + object);
```

```
        // Removing elements from the linked list
```

```
        object.remove("B");
```

```
        object.remove(3);
```

```
        object.removeFirst();
```

```
        object.removeLast();
```

```
        System.out.println("Linked list after deletion: " + object);
```

```
        // Finding elements in the linked list
```

```
        boolean status = object.contains("E");
```

```

        if(status)
            System.out.println("List contains the element 'E'
");
        else
            System.out.println("List doesn't contain the
element 'E'");

        // Number of elements in the linked list
        int size = object.size();
        System.out.println("Size of linked list = " + size);

        // Get and set elements from linked list
        Object element = object.get(2);
        System.out.println("Element returned by get() : " +
element);
        object.set(2, "Y");
        System.out.println("Linked list after change : " +
object);
    }
}

```

OUTPUT: Linked list : [D, A, E, B, C, F, G]
 Linked list after deletion: [A, E, F]
 List contains the element 'E'
 Size of linked list = 3
 Element returned by get() : F
 Linked list after change : [A, E, Y]