```
#include <iostream>
using namespace std;

struct Node {
   int data;
   Node* next;

};

class LinkedList {
private:
   Node* head=NULL;
public:

   // Insert at the end
   void Insert(int val) {
      Node* newNode = new Node;
      newNode->data=val;
      newNode->next = NULL;

      if (head == NULL) {
         head = newNode;

      }
      else{
      Node* current = head;
      while (current->next != NULL) {
         current = current->next;
      }
      current->next = newNode;
   }
   }

   // Display the list
   void display() {
      Node* current = head;
      while (current != nullptr) {
         cout << current->data <<  endl;
         current = current->next;
      }
      cout << endl;
   }
```

```
bool Search(int a)
{
   Node*curr=head;
   while(curr!=NULL)
   {
      if(curr->data==a)
      {
         return true;
      }
      else
      {
         curr=curr->next;
      }

   }
   return false;
}


void deleteNode(int a) {

Node* curr = head;
Node* prev = NULL;

while(curr!= NULL && curr->data!=a)
{
   prev=curr;
   curr= curr->next;
}

if(curr->data==a)
{
   if(curr->next==NULL)
   {
      prev->next=NULL;
      delete curr;
   }
   else if(curr == head) {
   head = head->next;
   delete curr;

}

else
```

```
    {
        prev->next= curr->next;
        delete curr;
    }

}
}




};

int main() {
    LinkedList list1;
    list1.Insert(1);
    list1.Insert(2);
    list1.Insert(3);
    list1.Insert(4);
    list1.Insert(5);
    list1.display();// Output: 1 2 3

//    cout<<list1.Search(5);
//    cout<<list1.Search(50);
//    list1.Search(50);

//    list1.deleteNode(5);
    list1.deleteNode(3);



    list1.display();

    return 0;
}
```