| Experiment No. 7 |
| --- |
| Program for data structure using built in function for link list, stack and queues |
| Date of Performace: 28/02/2024 |
| Date of Submission: 13/03/2024 |





             





**Code:**

l=[]

n=int(input("Enter number of elements to put in LL: "))

for i in range(n):

l.append(int(input()))

print("Printing index of 2")

if 2 in l:

print(l.index(2))

else:

print("Element 2 not found")

print("Inserting 45 at index 3")

l.insert(3,45)

for element in l:

print(element)

print("Removing 2")

if 2 in l:

l.remove(2)

else:

print("Element 2 not found")

for element in l:

print(element)

print(f'Size of list: {len(l)}')

search\_element = int(input("Enter the element to search for its location: "))

if search\_element in l:

print(f"Location of {search\_element} in the list: {l.index(search\_element)}")

else:

print(f"{search\_element} not found in the list.")

**Output:**

Enter number of elements to put in LL: 5

1

2

3

4

5

Printing index of 2

1

Inserting 45 at index 3

1

2

3

45

4

5

Removing 2

1

3

45

4

5

Size of list: 5

Enter the element to search for its location: 3

Location of 3 in the list: 1

**Conclusion:**

Linked lists provide a dynamic and efficient data structure for storing and manipulating data. They offer benefits such as dynamic memory allocation, efficient insertion and deletion operations, and non-contiguous memory allocation, making them suitable for scenarios where frequent modifications to the data structure are expected.