LAB REPORT

CSE 114 : Data Structure and Algorithms Sessional

PREPARED BY

Mehrin Farzana
ID: 2101013
Session: 2021-2022
Date: 23/08/2023

SUPERVISED BY

Suman Saha
Lecturer
Department of IRE, BDU

BANGABANDHU SHEIKH MUJIBUR

RAHMAN DIGITAL UNIVERSITY

(BDU)

List of Problems

1. Suppose you are given a sorted linked list. Take input from the user and insert it into the linked list maintaining the particular order.

2. Suppose you are given a sorted linked list. Take input from the user. If the input matches with any item on the list, delete it otherwise print the item not found.

Problem No.: 01

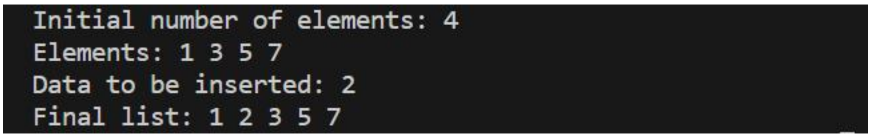
Problem Statement:

Suppose you are given a sorted linked list. Take input from the user and insert it into the linked list maintaining the particular order.

Code:

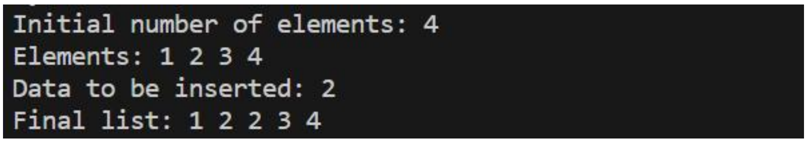
```
#include <stdio.h>
#include <stdlib.h>

struct node{
   int data;
   struct node *next;
};

int main() {
   int n,i;
   printf("Initial number of elements: ");
   scanf("%d", &n);
   struct node a[n], *j, *k, input;
   printf("Elements: ");
   for(i=0; i<n; i++){
      scanf("%d", &a[i].data);
      if(i>0){
         a[i-1].next=&a[i];
      }
   }
   a[i-1].next=NULL;
   printf("Data to be inserted: ");
   k=&a[0];
   scanf("%d", &input.data);
   for(j=k; j!=NULL; j=j->next){
      if(input.data>=j->data && input.data<=j->next->data){
         input.next=j->next;
         j->next=&input;
         break;
      }
      else if(input.data<a[0].data){
         input.next=k;
         k=&input;
         break;
      }
      else if(input.data>a[n-1].data){
            a[n-1].next=&input;
            input.next=NULL;
            break;
        }
    }
    printf("Final list: ");
    for(j=k; j!=NULL; j=j->next){
        printf("%d ", j->data);
    }
    return 0;
}
```

Output:



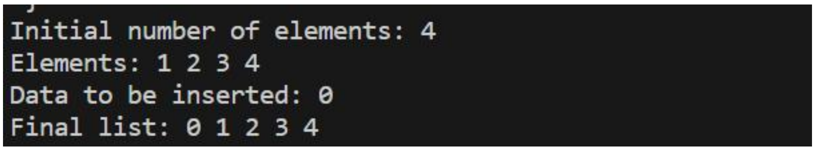
```
Initial number of elements: 4
Elements: 1 3 5 7
Data to be inserted: 2
Final list: 1 2 3 5 7
```

Fig 1.1: Output on console for case 1.



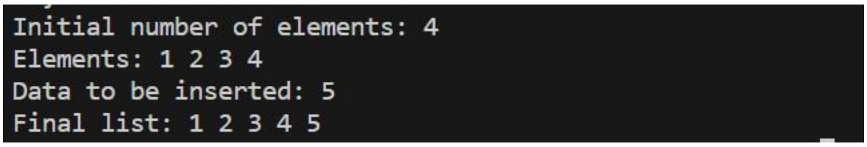
```
Initial number of elements: 4
Elements: 1 2 3 4
Data to be inserted: 2
Final list: 1 2 2 3 4
```

Fig 1.2: Output on console for case 2.



```
Initial number of elements: 4
Elements: 1 2 3 4
Data to be inserted: 0
Final list: 0 1 2 3 4
```

Fig 1.3: Output on console for case 3.



```
Initial number of elements: 4
Elements: 1 2 3 4
Data to be inserted: 5
Final list: 1 2 3 4 5
```

Fig 1.4: Output on console for case 4.

Problem No.: 02

Problem Statement:

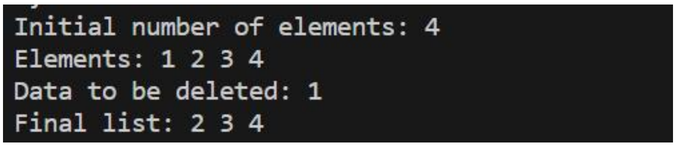
Suppose you are given a sorted linked list. Take input from the user. If the input matches with any item on the list, delete it otherwise print the item not found.

Code:

```
#include <stdio.h>
#include <stdlib.h>

struct node{
   int data;
   struct node *next;
};

int main() {
   int n,i, input,flag=0;
   printf("Initial number of elements: ");
   scanf("%d", &n);
   struct node a[n], *j, *k;
   printf("Elements: ");
   for(i=0; i<n; i++){
      scanf("%d", &a[i].data);
      if(i>0){
         a[i-1].next=&a[i];
      }
   }
   a[i-1].next=NULL;
   printf("Data to be deleted: ");
   k=&a[0];
   scanf("%d", &input);
   for(j=k; j->next!=NULL; j=j->next){
      if(input==j->next->data){
         j->next=j->next->next;
         flag=1;
         break;
      }
      else if(input==a[0].data){
         k=a[0].next;
         flag=1;
         break;
      }
   }
```

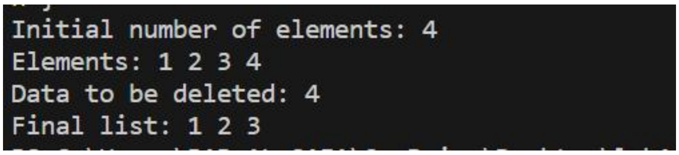
```
    if(!flag)
        printf("Item not found.\n");
    printf("Final list: ");
    for(j=k; j!=NULL; j=j->next){
        printf("%d ", j->data);
    }
    return 0;
}
```

Output:



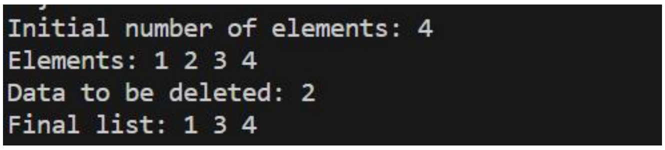
```
Initial number of elements: 4
Elements: 1 2 3 4
Data to be deleted: 1
Final list: 2 3 4
```

Fig 1.1: Output on console for case 1.



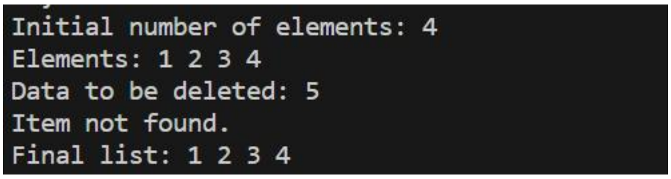
```
Initial number of elements: 4
Elements: 1 2 3 4
Data to be deleted: 4
Final list: 1 2 3
```

Fig 1.2: Output on console for case 2.



```
Initial number of elements: 4
Elements: 1 2 3 4
Data to be deleted: 2
Final list: 1 3 4
```

Fig 1.3: Output on console for case 3.



```
Initial number of elements: 4
Elements: 1 2 3 4
Data to be deleted: 5
Item not found.
Final list: 1 2 3 4
```

Fig 1.4: Output on console for case 4.