

```

1  #include<bits/stdc++.h>
2  using namespace std;
3  class node{
4      public:
5      int info;
6      node *next;
7      node *prev;
8  };
9  void creation(node **head,int data)
10 {
11     node *ptr=new node();
12     ptr->info=data;
13     ptr->next=NULL;
14     ptr->prev=NULL;
15     if((*head)==NULL)
16     {
17         *head=ptr;
18     }
19     else{
20         node *temp=*head;
21         while(temp->next!=NULL)
22         {
23             temp=temp->next;
24         }
25         temp->next=ptr;
26         ptr->prev=temp;
27     }
28 }
29
30 void first_insertion(node **head,int data)
31 {
32     node *ptr=new node;
33     ptr->info=data;
34     ptr->prev=NULL;
35     ptr->next=*head;
36     *head=ptr;
37 }
38 void last_insertion(node **head,int data)
39 {
40     node *ptr=new node();
41     ptr->info=data;
42     ptr->next=NULL;
43     ptr->prev=NULL;
44     node *temp=*head;
45     while(temp->next!=NULL)
46     {
47         temp=temp->next;
48     }
49     temp->next=ptr;
50     ptr->prev=temp;
51 }
52 int after_insertion(node **head,int data,int sear,int n){
53     node *ptr=new node;
54     ptr->info=data;
55     node *temp=*head;
56     while(temp!=NULL)
57     {
58         if(sear==temp->info)
59         {
60             if(temp->next==NULL)
61             {
62                 temp->next=ptr;
63                 ptr->prev=temp;
64                 ptr->next=NULL;
65                 return n+1;
66             }

```

```

67         else
68         {
69             ptr->next=temp->next;
70             ptr->prev=temp;
71             temp->next=ptr;
72             return n+1;
73         }
74     }
75     else
76     {
77         temp=temp->next;
78     }
79 }
80 if(temp==NULL){
81     cout<<"NOT FOUND"<<endl;
82 }
83 }
84 void before_insertion(node **head,int data,int sear)
85 {
86     node *ptr=new node();
87     node *preptr=new node();
88     ptr->info=data;
89     node *temp=*head;
90     if(temp->info==sear){
91         ptr->next=*head;
92         *head=ptr;
93     }
94     else{
95         while(temp->info!=sear){
96             preptr=temp;
97             temp=temp->next;
98         }
99         ptr->prev=preptr;
100        preptr->next=ptr;
101        ptr->next=temp;
102        temp->prev=ptr;
103    }
104 }
105 void insertPosition(node **head,int data,int pos)
106 {
107     node *ptr=new node;
108     ptr->info=data;
109     node *temp=*head;
110     if(pos==1)
111     {
112         ptr->next=*head;
113         *head=ptr;
114     }
115     else{
116         while(--pos>1)
117         {
118             temp=temp->next;
119         }
120         ptr->next=temp->next;
121         temp->next=ptr;
122         ptr->prev=temp;
123     }}
124 void print(node **head)
125 {
126     node *temp=*head;
127     while(temp!=NULL)
128     {
129         cout<<temp->info<<" ";
130         temp=temp->next;
131     }
132     cout<<endl;

```

```

133
134 }
135 int main()
136 {
137     node *head=new node();
138     head=NULL;
139     int data,i,n,val,pos;
140     cin>>n;
141     for(i=0;i<n;i++)
142     {
143         cin>>data;
144         creation(&head,data);
145     }
146     print(&head);
147     /*cin>>data;
148     first_insertion(&head,data);
149     print(&head);
150     cin>>data;
151     last_insertion(&head,data);
152     print(&head);
153     cin>>data >> val;
154     n=after_insertion(&head,data,val,n);
155     print(&head);
156     cin>>data >>val;
157     before_insertion(&head,data,val);
158     print(&head);*/
159     cin>>data >> pos;
160     insertPosition(&head,data,pos);
161     print(&head);
162 }

```