

Aim:**Source Code:**SingleLL5.c

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

#include "DelAtBegin.c"

void main() {
    NODE first = NULL;
    int x, op;
    while(1) {
        printf("1.Insert At Begin 2.Delete at Begin 3.Traverse the List 4.Exit\n");
        printf("Enter your option : ");
        scanf("%d", &op);
        switch(op) {
            case 1: printf("Enter an element : ");
                    scanf("%d", &x);
                    first = insertAtBegin(first, x);
                    break;
            case 2: if (first == NULL) {
                        printf("Single Linked List is empty so deletion is not possible\n");
                    } else {
                        first = deleteAtBegin(first);
                    }
                    break;
            case 3: if (first == NULL) {
                        printf("Single Linked List is empty\n");
                    } else {
                        printf("The elements in SLL are : ");
                        traverseList(first);
                    }
                    break;
            case 4: exit(0);
        }
    }
}
```

DelAtBegin.c

```
struct node {
    int data;
    struct node *next;
};
typedef struct node *NODE;

NODE createNode() {
```

```

    NODE temp;
    temp = (NODE) malloc(sizeof(struct node));
    temp -> next = NULL;
    return temp;
}

NODE insertAtBegin(NODE first, int x) {
    NODE temp;
    temp=createNode();
    temp->data=x;
    temp->next=first;
    first=temp;
    return first;
}

NODE deleteAtBegin(NODE first) {
    if(first==NULL)
    {
        printf("List is empty");
    }
    else
    {
        NODE temp=first;
        first=first->next;
        printf("The deleted element from SLL : %d\n",temp->data);
        free(temp);
        return first;
    }
}

void traverseList(NODE first) {
    NODE temp = first;
    while (temp != NULL) {
        printf("%d --> ",temp -> data);
        temp = temp -> next;
    }
    printf("NULL\n");
}

```

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
1.Insert At Begin 2.Delete at Begin 3.Traverse the List 4.Exit 1
Enter your option : 1
Enter an element : 10
1.Insert At Begin 2.Delete at Begin 3.Traverse the List 4.Exit 3
Enter your option : 3
The elements in SLL are : 10 --> NULL 2
1.Insert At Begin 2.Delete at Begin 3.Traverse the List 4.Exit 2
Enter your option : 2
The deleted element from SLL : 10 3
1.Insert At Begin 2.Delete at Begin 3.Traverse the List 4.Exit 3

Enter your option : 3
Single Linked List is empty 4
1.Insert At Begin 2.Delete at Begin 3.Traverse the List 4.Exit 4
Enter your option : 4