1.implement stack using singly linked list.

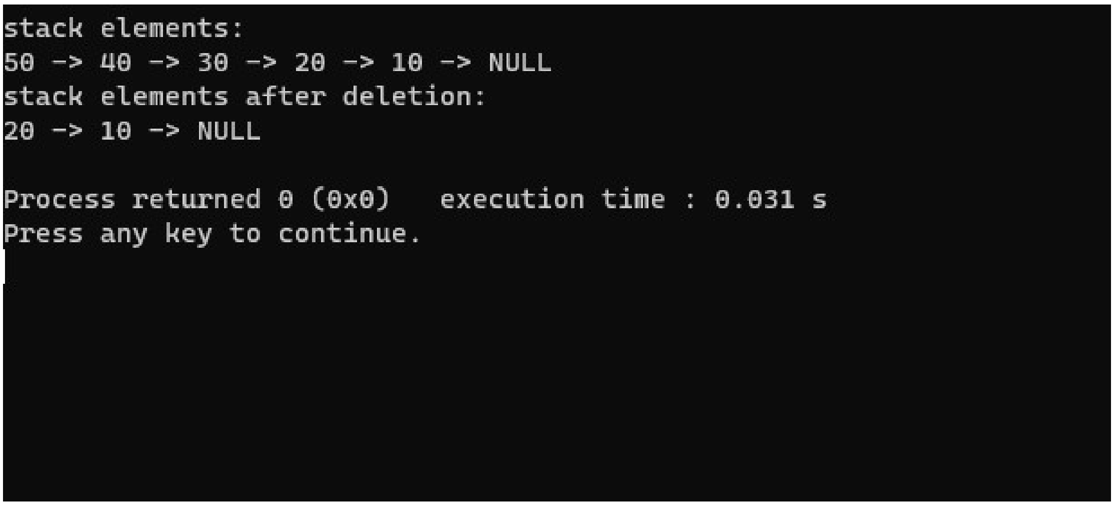
Shashank Patel C J

1BM22CS255

```
#include <stdio.h>

#include <stdlib.h>


struct Node {

   int data;

   struct Node* next;

};


void insertAtBeginning(struct Node** head, int value) {

   struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));

   newNode->data = value;

   newNode->next = *head;

   *head = newNode;

}

void deleteAtBeginning(struct Node** head) {

   if (*head == NULL) {

      printf("Linked list is already empty.\n");

      return;

   }

   struct Node* temp = *head;

   *head = (*head)->next;

   free(temp);
```

```
}
void display(struct Node* head)
{
    struct Node* temp = head;

    if (temp == NULL) {
        printf("Linked list is empty.\n");
        return;
    }

    while (temp != NULL) {
        printf("%d -> ", temp->data);
        temp = temp->next;
    }

    printf("NULL\n");
}
int main()
{
    struct Node* head=NULL;
    insertAtBeginning(&head,10);
    insertAtBeginning(&head,20);
    insertAtBeginning(&head,30);
    insertAtBeginning(&head,40);
    insertAtBeginning(&head,50);
    printf("stack elements:\n");
    display(head);

    deleteAtBeginning(&head);

    deleteAtBeginning(&head);

    deleteAtBeginning(&head);

    printf("stack elements after deletion:\n");

    display(head);

    return 0;

}
```

Output:



```
stack elements:
50 -> 40 -> 30 -> 20 -> 10 -> NULL
stack elements after deletion:
20 -> 10 -> NULL

Process returned 0 (0x0)   execution time : 0.031 s
Press any key to continue.
```