DSA Assignment: 6

Exp 6: Linked List implementation of Stack & Queue

Shashwat Tripathi

D10A Roll No: 60

AIM: In this experiment, we will perform Linked List implementation of Stack

CODE:

```c
// Exp 06 Linked List implementation of Stack
#include <stdio.h>
#include <stdlib.h>
struct Node
{
    int data;
    struct Node *next;
};
struct Node *top = NULL;

void push(int value);
int pop();
void display();
int main()
{
    int choice, value;
    printf("D10A_60_Shashwat Tripathi");
    printf("\nImplementaion of Stack using Linked List\n");

        printf("\n##############################################\n");
        printf("1. Push\n2. Pop\n3. Display\n4.Exit\n");
        printf("\n##############################################\n");

    while (1)
    {   printf("\nEnter your choice : ");
        scanf("%d", &choice);
        switch (choice) {
        case 1:
            printf("\nEnter the value to insert: ");
            scanf("%d", &value);
            push(value);
            break;
        case 2:
            printf("Popped element is :%d\n", pop());
            break;
        case 3:
            display();
            break;
        case 4:
            exit(0);
            break;
        default:
            printf("\nWrong Choice\n");
        }
```

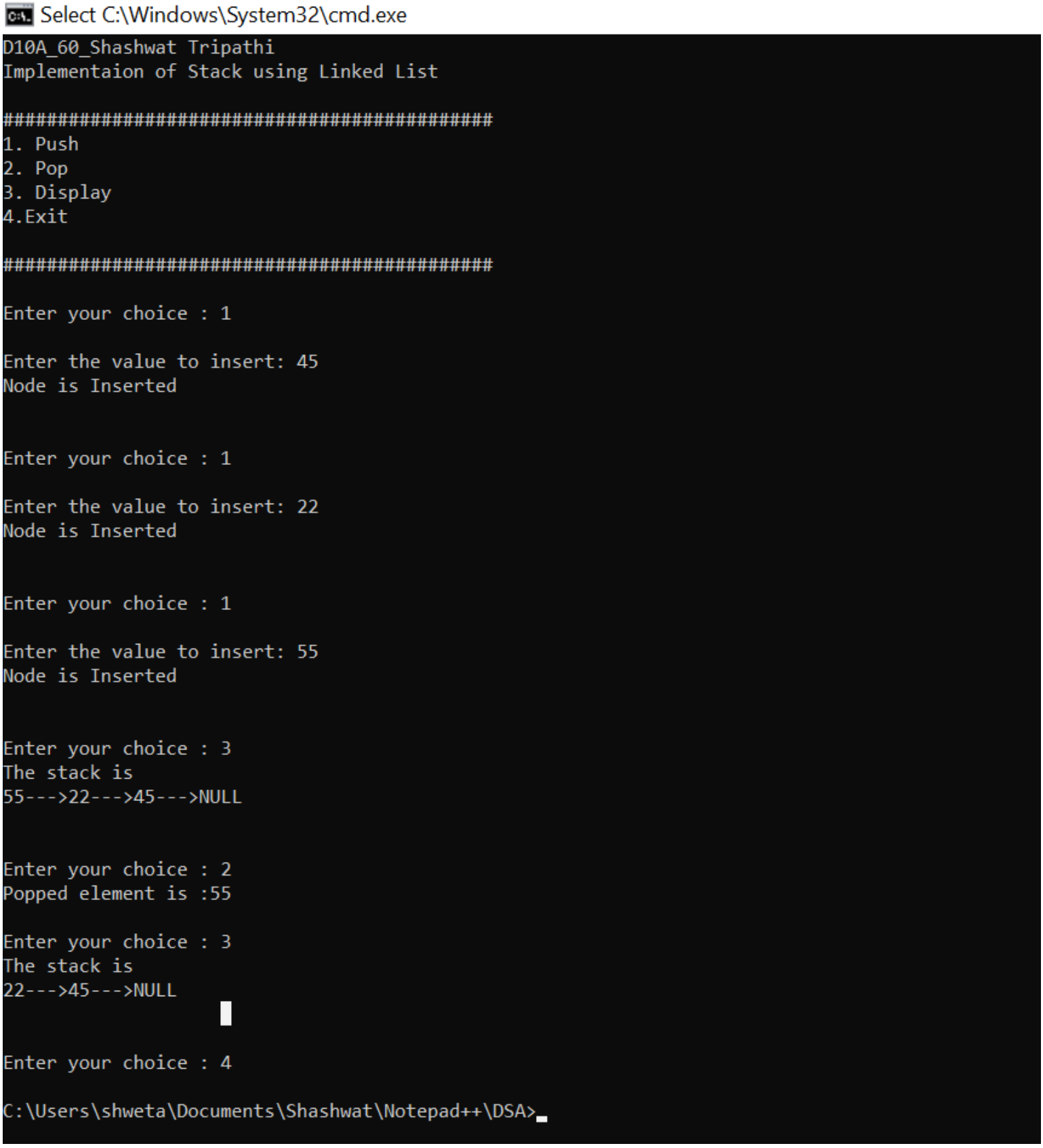
```
    }
}

void push(int value)
{
    struct Node *newNode;
    newNode = (struct Node*)malloc(sizeof(struct Node));
    newNode->data = value;
    if (top == NULL)
    {
        newNode->next = NULL;
    }
    else
    {
        newNode->next = top;
    }
    top = newNode;
    printf("Node is Inserted\n\n");
}

int pop()
{
    if (top == NULL)
    {
        printf("\nStack Underflow\n");
    }
    else
    {
        struct Node *temp = top;
        int temp_data = top->data;
        top = top->next;
        free(temp);
        return temp_data;
    }
}

void display()
{
    if (top == NULL)
    {
        printf("\nStack Underflow\n");
    }
    else
    {
        printf("The stack is \n");
        struct Node *temp = top;
        while (temp->next != NULL)
        {
            printf("%d--->", temp->data);
            temp = temp->next;
        }
        printf("%d--->NULL\n\n", temp->data);
    }
}
```

OUTPUT:


Select C:\Windows\System32\cmd.exe
D10A_60_Shashwat Tripathi
Implementaion of Stack using Linked List
##
1. Push
2. Pop
3. Display
4.Exit
##
Enter your choice : 1
Enter the value to insert: 45
Node is Inserted
Enter your choice : 1
Enter the value to insert: 22
Node is Inserted
Enter your choice : 1
Enter the value to insert: 55
Node is Inserted
Enter your choice : 3
The stack is
55--->22--->45--->NULL
Enter your choice : 2
Popped element is :55
Enter your choice : 3
The stack is
22--->45--->NULL
Enter your choice : 4
C:\Users\shweta\Documents\Shashwat\Notepad++\DSA>

AIM: In this experiment, we will perform Linked List implementation of Queue

CODE:

```
// Exp 06 Linked List implementation of Queue
#include <stdio.h>
#include <stdlib.h>
struct node
{
    int data;
    struct node *next;
};
struct node *front;
struct node *rear;
void insert();
void delete ();
void display();
void main()
{
    printf("D10A_60_Shashwat Tripathi");
    int choice;
    printf("\n################################################\n");
    printf("\n1.insert an element\n2.Delete an element\n3.Display the
queue\n4.Exit\n");
    printf("\n################################################\n");
    while(choice != 4)
    {
        printf("\nEnter your choice:");
        scanf("%d", &choice);
        switch (choice)
        {
        case 1:
            insert();
            break;
        case 2:
            delete ();
            break;
        case 3:
            display();
            break;
        case 4:
            exit(0);
            break;
        default:
            printf("\nEnter valid choice??\n");
        }
    }
}
```

```
void insert()
{
    struct node *ptr;
    int item;
    ptr = (struct node *)malloc(sizeof(struct node));
    if (ptr == NULL)
    {
        printf("\nOVERFLOW\n");
        return;
    }
    else
    {
        printf("\nEnter value?\n");
        scanf("%d", &item);
        printf("Element is inserted\n");
        ptr->data = item;
        if (front == NULL)
        {
            front = ptr;
            rear = ptr;
            front->next = NULL;
            rear->next = NULL;
        }
        else
        {
            rear->next = ptr;
            rear = ptr;
            rear->next = NULL;
        }
    }
}
void delete ()
{
    struct node *ptr;
    if (front == NULL)
    {
        printf("\nUNDERFLOW\n");
        return;
    }
    else
    {
        printf("Element is Deleted\n");
        ptr = front;
        front = front->next;
        free(ptr);
    }
}
void display()
{
    struct node *ptr;
    ptr = front;
    if (front == NULL)
    {
        printf("\nEmpty queue\n");
```

```
        }
        else
        {
            printf("\nprinting values .....\n");
            while (ptr != NULL)
            {
                printf("%d\n",ptr ->data);
                ptr = ptr->next;
            }
        }
    }
```

OUTPUT:

Select C:\Windows\System32\cmd.exe

```
C:\Users\shweta\Documents\Shashwat\Notepad++\DSA>DSAexp6
D10A_60_Shashwat Tripathi
##########################################

1.insert an element
2.Delete an element
3.Display the queue
4.Exit

##########################################

Enter your choice:1

Enter value?
61
Element is inserted

Enter your choice:1

Enter value?
43
Element is inserted

Enter your choice:1

Enter value?
55
Element is inserted

Enter your choice:3

printing values .....
61
43
55

Enter your choice:2
Element is Deleted

Enter your choice:3

printing values .....
43
55

Enter your choice:4

C:\Users\shweta\Documents\Shashwat\Notepad++\DSA>
```