



Academic Year: 2022-2023

**Name:** Nikki Mehta

**Sap Id:** 60018220019/S040

**Branch:** Artificial Intelligence and Data Science

**Div:** S

**Course:** Data Structures

### Experiment No – 4

**AIM:** To implement Stack and Queue using Linked List

**CODE:** 1] Stack Using Linked List

```
#include <stdio.h>
#include<stdlib.h>
void push(int);
int pop();
void display();
struct node
{
    int data;
    struct node* next;
}*top=NULL;

void push(int data)
{
    struct node* newNode=(struct node*)malloc(sizeof(struct node));
    newNode->data=data;
    newNode->next=top;
    top=newNode;
}

int pop()
{
    struct node *temp;
    temp=top;
    int value=temp->data;
    top=temp->next;
    free(temp);
    return value;
}

int stackTop()
{
    return top->data;
}

void display()
{
```



Academic Year: 2022-2023

```
struct node *temp;
temp=top;
while(temp!=NULL)
{
    printf("%d->",temp->data);
    temp=temp->next;
}
}

int isEmpty()
{
    return top == NULL;
}

void main()
{
    int choice, data;

    do
    {
        printf("\nAI-DS-60018220019");
        printf("\nStack Operations:\n");
        printf("1. Push\n");
        printf("2. Pop\n");
        printf("3. Peek (Stack Top)\n");
        printf("4. Display\n");
        printf("5. Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch (choice)
        {
            case 1:
                printf("Enter data to push: ");
                scanf("%d", &data);
                push(data);
                display();
                break;
            case 2:
                if (!isEmpty())
                {
                    printf("Popped element: %d\n", pop());
                    display();
                }
                else
                {
                    printf("Stack is empty. Cannot pop from an empty stack.\n");
                }
            }
        }
    }
```



Academic Year: 2022-2023

```
        break;
    case 3:
        if (!isEmpty())
            printf("Stack Top: %d\n", stackTop());
        break;
    case 4:
        printf("Stack elements: ");
        display();
        break;
    case 5:
        printf("Exiting The Program!");
        exit(0);
    default:
        printf("Invalid choice. Please enter a valid option.\n");
    }
}
while(choice!=5);
}
```

#### OUTPUT:

**AI-DS-60018220019**

**Stack Operations:**

- 1. Push**
- 2. Pop**
- 3. Peek (Stack Top)**
- 4. Display**
- 5. Exit**

**Enter your choice: 1**

**Enter data to push: 5**

**5->**

**AI-DS-60018220019**

**Stack Operations:**

- 1. Push**
- 2. Pop**
- 3. Peek (Stack Top)**
- 4. Display**
- 5. Exit**

**Enter your choice: 1**

**Enter data to push: 6**

**6->5->**

**AI-DS-60018220019**

**Stack Operations:**

- 1. Push**
- 2. Pop**
- 3. Peek (Stack Top)**
- 4. Display**
- 5. Exit**

**Enter your choice: 1**

**Enter data to push: 7**

**7->6->5->**

**AI-DS-60018220019**

**Stack Operations:**

- 1. Push**
- 2. Pop**
- 3. Peek (Stack Top)**
- 4. Display**
- 5. Exit**

**Enter your choice: 2**

**Popped element: 7**

**6->5->**



Shri Vile Parle Kelavani Mandal's

**DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING**

(Autonomous College Affiliated to the University of Mumbai)

NAAC Accredited with "A" Grade (CGPA : 3.18)



Academic Year: 2022-2023

---

**AI-DS-60018220019**

**Stack Operations:**

1. Push
2. Pop
3. Peek (Stack Top)
4. Display
5. Exit

**Enter your choice: 3**

**Stack Top: 6**

**AI-DS-60018220019**

**Stack Operations:**

1. Push
2. Pop
3. Peek (Stack Top)
4. Display
5. Exit

**Enter your choice: 4**

**Stack elements: 6->5->**

**AI-DS-60018220019**

**Stack Operations:**

1. Push
2. Pop
3. Peek (Stack Top)
4. Display
5. Exit

**Enter your choice: 5**

**Exiting The Program!**



## 2] Queue Using Linked List

### CODE:-

```
#include <stdio.h>
#include <stdlib.h>
void enqueue(int );
void dequeue();
void queueFront();
void queueRear();
void display();
struct node
{
    int data;
    struct node* next;
};
struct node *rear=NULL,*front=NULL,*newNode,*temp;
void main()
{
    int choice,data;
    do
    {
        printf("\nAI-DS-60018220019");
        printf("\n*Queue Operations*\n");
        printf("\n1.Enqueue\n2.Dequeue\n3.QueueFront\n4.QueueRear\n5.Exit\n");
        printf("Enter choice\n");
        scanf("%d",&choice);
        switch(choice)
        {
            case 1:printf("Enter data");
                    scanf("%d",&data);
                    enqueue(data);
                    display();
                    break;
            case 2:dequeue();
                    display();
                    break;

            case 3: queueFront();
                    break;
            case 4: queueRear();
                    break;
            case 5: printf("Exiting program");
                    exit(0);
                    break;
            default:printf("Enter valid choice");
```



Academic Year: 2022-2023

```
}  
}  
while(choice!=5);  
}  
  
struct node* createNode(int info)  
{  
    struct node* newNode=(struct node*)malloc(sizeof(struct node));  
    newNode->data=info;  
    newNode->next=NULL;  
    return newNode;  
}  
void enqueue(int info)  
{  
    newNode=createNode(info);  
    if(front==NULL && rear==NULL)  
    {  
        front=newNode;  
        rear=newNode;  
    }  
    else  
    {  
        rear->next=newNode;  
        rear=newNode;  
    }  
}  
void dequeue()  
{  
    if(front==NULL)  
    {  
        printf("\n Queue is empty");  
    }  
    else  
    {  
        temp=front;  
        printf("\nDequeued element is %d\n",front->data);  
        front=front->next;  
        free(temp);  
    }  
}  
void queueFront()  
{  
    if(front==NULL)  
    {  
        printf("\n Queue is empty");  
    }  
    else
```



Academic Year: 2022-2023

```
{
    printf("Queue Front is %d ",front->data);

}
}
void queueRear()
{
    if(front==NULL)
    {
        printf("\n Queue is empty");
    }
    else
    {
        printf("Queue Rear is %d ",rear->data);
    }
}
void display()
{
    if(front==NULL)
    {
        printf("\n Queue is empty");
    }
    else
    {
        printf("\nQueue is: ");
        temp=front;
        while(temp!=NULL)
        {
            printf("%d-->",temp->data);
            temp=temp->next;
        }
    }
}
```



Academic Year: 2022-2023

---

**OUTPUT:-**

**AI-DS-60018220019**

**\*Queue Operations\***

- 1.Enqueue
- 2.Dequeue
- 3.QueueFront
- 4.QueueRear
- 5.Exit

Enter choice

1

Enter data5

Queue is: 5-->

**AI-DS-60018220019**

**\*Queue Operations\***

- 1.Enqueue
- 2.Dequeue
- 3.QueueFront
- 4.QueueRear
- 5.Exit

Enter choice

1

Enter data3

Queue is: 5-->3-->

**AI-DS-60018220019**

**\*Queue Operations\***

- 1.Enqueue
- 2.Dequeue
- 3.QueueFront
- 4.QueueRear
- 5.Exit

Enter choice

1

Enter data6

Queue is: 5-->3-->6-->

**AI-DS-60018220019**

**\*Queue Operations\***

- 1.Enqueue
- 2.Dequeue
- 3.QueueFront
- 4.QueueRear
- 5.Exit

Enter choice

2

Dequeued element is 5

Queue is: 3-->6-->

**AI-DS-60018220019**

**\*Queue Operations\***

- 1.Enqueue
- 2.Dequeue
- 3.QueueFront
- 4.QueueRear
- 5.Exit

Enter choice

3

Queue Front is 3

**AI-DS-60018220019**

**\*Queue Operations\***

- 1.Enqueue
- 2.Dequeue
- 3.QueueFront
- 4.QueueRear
- 5.Exit

Enter choice

4

Queue Rear is 6