Assignment No. 5

Program:-

```
#include <iostream>
using namespace std;
#define SIZE 5
class dequeue
{
	int a[10], front, rear, count;

public:
	dequeue();
	void add_at_beg(int);
	void add_at_end(int);
	void delete_fr_front();
	void delete_fr_rear();
	void display();
};
dequeue::dequeue()
{
	front =
		-1;
	rear =
		-1;
	count = 0;
}
void dequeue::add_at_beg(int item)
{
	int i;
```

```
        if (front ==
                -1)
        {
                front++;
                rear++;
                a[rear] = item;
                count++;
        }
        else if (rear >= SIZE - 1)
        {
        }
        else
        {
                cout << "\nInsertion is not possible,overflow!!!!";
                for (i = count; i >= 0; i--)
                {
                        a[i] = a[i - 1];
                }
                a[i] = item;
                count++;
                rear++;
        }
}
void dequeue::add_at_end(int item)
{
        if (front == -1)
        {
                front++;
                rear++;
```

```
                a[rear] = item;
                count++;
        }
        else if (rear >= SIZE - 1)
        {
                cout << "\nInsertion is not possible,overflow!!!";
                return;
        }
        else
        {
                a[++rear] = item;
        }
}
void dequeue::display()
{
        for (int i = front; i <= rear; i++)
        {
                cout << a[i] << " ";
        }
}
void dequeue::delete_fr_front()
{
        if (front == -1)
        {
        }
        else
        {
                cout << "Deletion is not possible:: Dequeue is empty";
                return;
```

```
                if (front == rear)
                {
                        front = rear = -1;
                        return;
                }
                cout << "The deleted element is " << a[front];
                front = front + 1;
        }
}
void dequeue::delete_fr_rear()
{
        if (front == -1)
        {
        }
        else
        {
                cout << "Deletion is not possible:Dequeue is empty";
                return;
                if (front == rear)
                {
                        front = rear = -1;
                }
                cout << "The deleted element is " << a[rear];
                rear = rear - 1;
        }
}
int main()
{
        int c, item;
```

```
dequeue d1;
do
{
        cout << "\n\n****DEQUEUE OPERATION****\n";
        cout << "\n1-Insert at beginning";
        cout << "\n2-Insert at end";
        cout << "\n3_Display";
        cout << "\n4_Deletion from front";
        cout << "\n5-Deletion from rear";
        cout << "\n6_Exit";
        cout << "\nEnter your choice<1-4>:";
        cin >> c;
        switch (c)
        {
        case 1:
                cout << "Enter the element to be inserted:";
                cin >> item;
                d1.add_at_beg(item);
                break;
        case 2:
                cout << "Enter the element to be inserted:";
                cin >> item;
                d1.add_at_end(item);
                break;
        case 3:
                d1.display();
                break;
        case 4:
                d1.delete_fr_front();
```

```
                break;
            case 5:
                d1.delete_fr_rear();
                break;
            case 6:
                exit(1);
                break;
            default:
                cout << "Invalid choice";
                break;
            }
        } while (c != 7);
        return 0;
}
```

Output :-

```
****DEQUEUE OPERATION****
1-Insert at beginning
2-Insert at end
3_Display
4_Deletion from front
5-Deletion from rear
6_Exit
Enter your choice<1-4>:1
Enter the element to be inserted:45
****DEQUEUE OPERATION****
1-Insert at beginning
2-Insert at end
3_Display
4_Deletion from front
```

5-Deletion from rear

6_Exit

Enter your choice<1-4>:2

Enter the element to be inserted:46