Data Structure Lab
CEN-391

Program 7

Code :-

```cpp
#include <iostream>
using namespace std;

void isEmpty(int size)
{
    cout << "isEmpty...\n";
    if (size == -1)
        cout << "Empty" << endl;
    else
        cout << "Not Empty" << endl;
}

void isFull(int size, int capacity)
{
    cout << "isFull...\n";
    if (size + 1 == capacity)
```

```cpp
            cout << "Full" << endl;
        else
            cout << "Not Full" << endl;
    }

    void Display(int queue[], int size)
    {
        cout << "Display...\n";
        if (size == -1)
        {
            cout << "Queue Is Empty" << endl;
            return;
        }
        for (int i = 0; i <= size; i++)
        {
            cout << queue[i] << " ";
        }
        cout << endl;
    }

    void Enqueue(int queue[], int &size, int capacity)
    {
        cout << "Enqueue...\n";
        size++;
        if (size == capacity)
        {
            size--;
            cout << "Queue Overflow" << endl;
            return;
        }
        cout << "Enter The Element : ";
        cin >> queue[size];
        Display(queue, size);
    }

    void Dequeue(int queue[], int &size)
    {
        cout << "Dequeue...\n";
        if (size == -1)
```

```cpp
        {
            cout << "Queue Underflow" << endl;
            return;
        }
        cout<<queue[0]<<endl;
        for (int i = 1; i <= size; i++)
        {
            queue[i - 1] = queue[i];
        }
        size--;
        Display(queue, size);
    }
    void Front_Rear(int queue[], int size)
    {
        cout << "Front And Rear...\n";
        if (size == -1)
        {
            cout << "Queue Is Empty" << endl;
            return;
        }
        cout << "Front : " << queue[0] << endl;
        cout << "Rear : " << queue[size] << endl;
    }

    void Total_Element(int size)
    {
        cout << "Total Elements In Queue : " << size + 1 <<
endl;
    }

    void Bars()
    {
        cout << "-------------------------------------------------
----------------\n";
    }
    bool Options(int queue[], int &size, int capacity)
    {
        int opt;
        cin >> opt;
```

```cpp
        Bars();
        switch (opt)
        {
        case 1:
            Enqueue(queue, size, capacity);
            break;
        case 2:
            Dequeue(queue, size);
            break;
        case 3:
            Front_Rear(queue, size);
            break;
        case 4:
            isEmpty(size);
            break;
        case 5:
            isFull(size, capacity);
            break;
        case 6:
            Total_Element(size);
            break;
        case 7:
            Display(queue, size);
            break;
        case 8:
            cout << "Exit...\n";
            return 0;
        default:
            cout << "Invalid Input!\nTry Again!\n";
        }
        Bars();
        return 1;
    }

    void Menu()
    {
        cout << "_____Operations_On_Queue_____ \n";
        cout << "1.Enqueue \n";
        cout << "2.Dequeue \n";
```

```cpp
        cout << "3.Front And Rear Element \n";
        cout << "4.isEmpty \n";
        cout << "5.isFull \n";
        cout << "6.Total Elements \n";
        cout << "7:Display \n";
        cout << "8.Exit \n";
        cout << "Enter Your Choice : ";
    }

    int main()
    {

        system("cls");
        cout << "_____Vicky_Gupta_20BCS070_____\n\n";
        cout << "Enter The Size Of The Queue : ";
        int capacity, size = -1;
        cin >> capacity;
        int queue[capacity] = {0};
        cout << "\n\n";
        while (true)
        {
            Menu();
            if (!Options(queue, size, capacity))
                break;
        }
        cout << "Exiting...\n";
        Bars();
        return 0;
    }
```

Output :-

```
_____Vicky_Gupta_20BCS070_____

Enter The Size Of The Queue : 3


_____Operations_On_Queue_____
1.Enqueue
2.Dequeue
3.Front And Rear Element
4.isEmpty
5.isFull
6.Total Elements
7:Display
8.Exit
Enter Your Choice : 1
---------------------------------
Enqueue...
Enter The Element : 11
Display...
11
---------------------------------
_____Operations_On_Queue_____
1.Enqueue
2.Dequeue
3.Front And Rear Element
4.isEmpty
5.isFull
6.Total Elements
7:Display
8.Exit
Enter Your Choice : 1
---------------------------------
Enqueue...
Enter The Element : 22
Display...
11 22
---------------------------------
```

```
_____Operations_On_Queue_____
1.Enqueue
2.Dequeue
3.Front And Rear Element
4.isEmpty
5.isFull
6.Total Elements
7:Display
8.Exit
Enter Your Choice : 1
-------------------------------------------
Enqueue...
Enter The Element : 33
Display...
11 22 33
-------------------------------------------
_____Operations_On_Queue_____
1.Enqueue
2.Dequeue
3.Front And Rear Element
4.isEmpty
5.isFull
6.Total Elements
7:Display
8.Exit
Enter Your Choice : 3
-------------------------------------------
Front And Rear...
Front : 11
Rear : 33
-------------------------------------------
_____Operations_On_Queue_____
1.Enqueue
2.Dequeue
3.Front And Rear Element
4.isEmpty
5.isFull
6.Total Elements
7:Display
8.Exit
Enter Your Choice : 5
-------------------------------------------
isFull...
Full
-------------------------------------------
```

```
_____Operations_On_Queue_____
1.Enqueue
2.Dequeue
3.Front And Rear Element
4.isEmpty
5.isFull
6.Total Elements
7:Display
8.Exit
Enter Your Choice : 6
----------------------------------------
Total Elements In Queue : 3
----------------------------------------
_____Operations_On_Queue_____
1.Enqueue
2.Dequeue
3.Front And Rear Element
4.isEmpty
5.isFull
6.Total Elements
7:Display
8.Exit
Enter Your Choice : 7
----------------------------------------
Display...
11 22 33
----------------------------------------
_____Operations_On_Queue_____
1.Enqueue
2.Dequeue
3.Front And Rear Element
4.isEmpty
5.isFull
6.Total Elements
7:Display
8.Exit
Enter Your Choice : 2
----------------------------------------
Dequeue...
11
Display...
22 33
----------------------------------------
```

```
_____Operations_On_Queue_____
1.Enqueue
2.Dequeue
3.Front And Rear Element
4.isEmpty
5.isFull
6.Total Elements
7:Display
8.Exit
Enter Your Choice : 2
----------------------------------
Dequeue...
22
Display...
33
----------------------------------
_____Operations_On_Queue_____
1.Enqueue
2.Dequeue
3.Front And Rear Element
4.isEmpty
5.isFull
6.Total Elements
7:Display
8.Exit
Enter Your Choice : 3
----------------------------------
Front And Rear...
Front : 33
Rear : 33
----------------------------------
_____Operations_On_Queue_____
1.Enqueue
2.Dequeue
3.Front And Rear Element
4.isEmpty
5.isFull
6.Total Elements
7:Display
8.Exit
Enter Your Choice : 2
```

```
_____Operations_On_Queue_____
1.Enqueue
2.Dequeue
3.Front And Rear Element
4.isEmpty
5.isFull
6.Total Elements
7:Display
8.Exit
Enter Your Choice : 2
------------------------------------------------
Dequeue...
33
Display...
Queue Is Empty
------------------------------------------------
_____Operations_On_Queue_____
1.Enqueue
2.Dequeue
3.Front And Rear Element
4.isEmpty
5.isFull
6.Total Elements
7:Display
8.Exit
Enter Your Choice : 4
------------------------------------------------
isEmpty...
Empty
------------------------------------------------
_____Operations_On_Queue_____
1.Enqueue
2.Dequeue
3.Front And Rear Element
4.isEmpty
5.isFull
6.Total Elements
7:Display
8.Exit
Enter Your Choice : 8
------------------------------------------------
Exit...
Exiting...
------------------------------------------------
```