DS PRACTICAL 08 [C]

AIM: Implement a Circular Queue and perform the Queue operations: Enqueue, Dequeue and Print using Menu Driver Program such as 1.Add, 2.Delete and 3.Print and 4.Exit.

PROGRAM:

```
#include <stdio.h>


// Creating array Globaly
int Queue[5];
int front = -1, rear = -1, data;


// FUNCTION FOR ENQUEUE
int enqueue()
{
   if((rear + 1) % 5 == front){
      printf("The Queue is Overflow.\n");
   }else if(front == -1 && rear == -1){
      front = 0;
      rear = 0;
      printf("Enter the data.\n");
      scanf("%d", &data);
      Queue[rear] = data;
   }else{
      printf("Enter the data.\n");
      scanf("%d", &data);
      rear = (rear + 1) % 5;
      Queue[rear] = data;
   }
   return 0;
}


// FUNCTION FOR DEQUEUE
```

```
int dequeue()

{

   if(front == -1 && rear == -1 ){
      printf("The Queue is Underflow.\n");
   }else if(front == rear){
      printf("The Queue is Underflow.\n");
      front = rear = -1;
   }else{
      printf("The deleting element is %d.\n", Queue[front]);
      front = (front + 1) % 5;
   }
   return 0;
}

void display()
{
   if (front == -1)
   {
      // Checking the queue is empty or not.
      printf("The Queue is empty so, can not print the element.\n");
   }
   else
   {
      // printing the elements in the Queue
      int i = front;
      while (1)
      {
         printf("%d\t", Queue[i]);
         if (i == rear)
```

```
            break;      // Stop when we reach the rear
        i = (i + 1) % 5; // Move to the next index in circular manner

    }
    printf("\n");
  }
}

// MAIN FUNCTION
int main()
{
  int choice;
  printf("Queue Implementation\n");
  printf("Choices\n1.Enqueue\t2.Dequeue\t3.Print\t4.Exit\n");
  do
  {
    printf("Enter a valid choice:");
    scanf("%d", &choice);

    switch (choice)
    {
    case 1:
      enqueue();
      break;
    case 2:
      dequeue();
      break;

    case 3:
      display();
      break;
```

```
        case 4:
            printf("You exited the Program successfully.");

            break;

        default:
            printf("Please enter a valid choice as mention!\n");
            break;
        }
    } while (choice != 4);

    return 0;
}
```

OUTPUT

```
PS C:\Users\chuna> g++ p8c.c
PS C:\Users\chuna> ./a.exe
Queue Implementation
Choices
1.Enqueue       2.Dequeue       3.Print 4.Exit
Enter a valid choice:1
Enter the data.
12
Enter a valid choice:1
Enter the data.
23
Enter a valid choice:1
Enter the data.
45
Enter a valid choice:2
The deleting element is 12.
Enter a valid choice:3
23      45
Enter a valid choice:4
You exited the Program successfully.
PS C:\Users\chuna>
```

GITHUB LINK : https://github.com/Nishikant-Chunarkar/DATA_STRUCTURE_PRACTICAL