PAGE No.
DATE / /

Circular Queue

```
#include <stdio.h>
#include <stdlib.h>
#define MAX 3

int front = -1;
int rear = -1;

int queue[MAX];
void Enque(int);
int Deque();
void display();
int main(int argc, char **argv)
{
    int option;
    int item;
    do{
        printf("Circular Queue\n");
        printf("\n1. Insert to Queque (Enqueue)");
        printf("\n2. delete from the Queue (Dequeue)");
        printf("\n3. Display the content");
        printf("\n4. Exit\n");
        printf("Enter the option: ");
        scanf("%d", &option);
        switch(option)
        {
            case 1: printf("Enter the element\n");
                    scanf("%d", &item);
                    Enque(item);
                    break;
            case 2: item = Deque();
                    if(item == -999)
                        printf("Queque is empty");
```

```
        else
        printf("Removed element %d from the queque", item);
        break;
case 3: display();
        break;
case 4: exit(0);
}
}while(option!=4)
return 0;
}
void Enque(int ele)
{
  if((front == 0 && rear == MAX-1) || (front == rear+1))
  {
    printf("Queue is full\n");
    return;
  }
  else
  {
    rear = (rear+1)%MAX;
    queue[rear] = ele;
    if(front == -1)
    {
      front=0;
    }
}
```

```
int Dequeue()
{
   int item;
   if((front==-1) && (rear==-1))
   {
     return(-999);
   }
   else
   {
      item = queue[front];
      if(front==rear)
      {
         front=-1;
         rear=-1;
      }
      else
      {
         front=(front+1)%MAX;
      }
      return item;
   }
}
void display()
{
   int i;
   if((front==-1) && (rear==-1) || front==rear))
   {
     printf("Queue is empty\n");
     return;
   }
```

```
else
{
    printf("\n Queue contents:\n");
    for(i = front; i<=rear; i++)
        printf("%d", queue[i]);
}
}
```