CIRCULAR QUEUE :

```
#include<iostream>

using namespace std;

class CircularQueue {
  private:
    int front;
  int rear;
  int arr[5];
  int itemCount;

  public:
    CircularQueue() {
      itemCount = 0;
      front = -1;
      rear = -1;
      for (int i = 0; i < 5; i++) {
        arr[i] = 0;
      }
    }
  bool isEmpty() {
    if (front == -1 && rear == -1)
      return true;
    else
      return false;
  }
  bool isFull() {
    if ((rear + 1) % 5 == front)
      return true;
    else
      return false;
  }
  void enqueue(int val) {
    if (isFull()) {
      cout << "Queue full" << endl;
      return;
    } else if (isEmpty()) {
      rear = 0;
      front = 0;
      arr[rear] = val;

    } else {
      rear = (rear + 1) % 5;
      arr[rear] = val;

    }
    itemCount++;

  }

  int dequeue() {
```

```
    int x = 0;
    if (isEmpty()) {
      cout << "Queue is Empty" << endl;
      return x;
    } else if (rear == front) {
      x = arr[rear];
      rear = -1;
      front = -1;
      itemCount--;
      return x;
    } else {
      cout << "front value: " << front << endl;
      x = arr[front];
      arr[front] = 0;
      front = (front + 1) % 5;
      itemCount--;
      return x;
    }
  }

  int count() {
    return (itemCount);
  }

  void display() {
    cout << "All values in the Queue are - " << endl;
    for (int i = 0; i < 5; i++) {
      cout << arr[i] << "  ";
    }
  }

};

int main() {
  CircularQueue q1;
  int value, option;

  do {
    cout << "\n\nWhat operation do you want to perform? Select
Option number. Enter 0 to exit." << endl;
    cout << "1. Enqueue()" << endl;
    cout << "2. Dequeue()" << endl;
    cout << "3. isEmpty()" << endl;
    cout << "4. isFull()" << endl;
    cout << "5. count()" << endl;
    cout << "6. display()" << endl;
    cout << "7. Clear Screen" << endl << endl;

    cin >> option;

    switch (option) {
    case 0:
      break;
    case 1:
```

```
      cout << "Enqueue Operation \nEnter an item to Enqueue in the
Queue" << endl;
      cin >> value;
      q1.enqueue(value);
      break;
    case 2:
      cout << "Dequeue Operation \nDequeued Value : " <<
q1.dequeue() << endl;
      break;
    case 3:
      if (q1.isEmpty())
        cout << "Queue is Empty" << endl;
      else
        cout << "Queue is not Empty" << endl;
      break;
    case 4:
      if (q1.isFull())
        cout << "Queue is Full" << endl;
      else
        cout << "Queue is not Full" << endl;
      break;
    case 5:
      cout << "Count Operation \nCount of items in Queue : " <<
q1.count() << endl;
      break;
    case 6:
      cout << "Display Function Called - " << endl;
      q1.display();
      break;
    case 7:
      system("cls");
      break;
    default:
      cout << "Enter Proper Option number " << endl;
    }

  } while (option != 0);

  return 0;
}
```