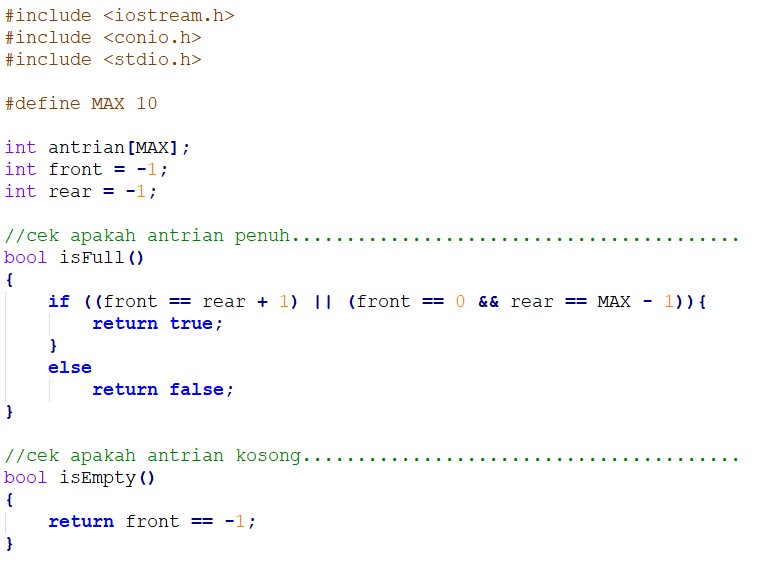
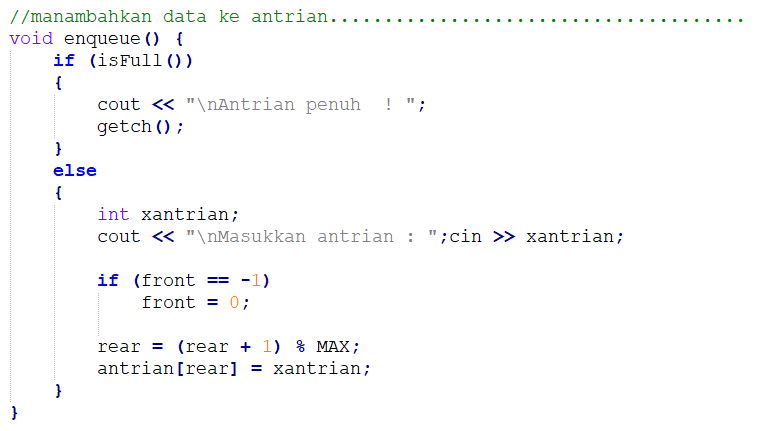
**Topik 5 Queue**

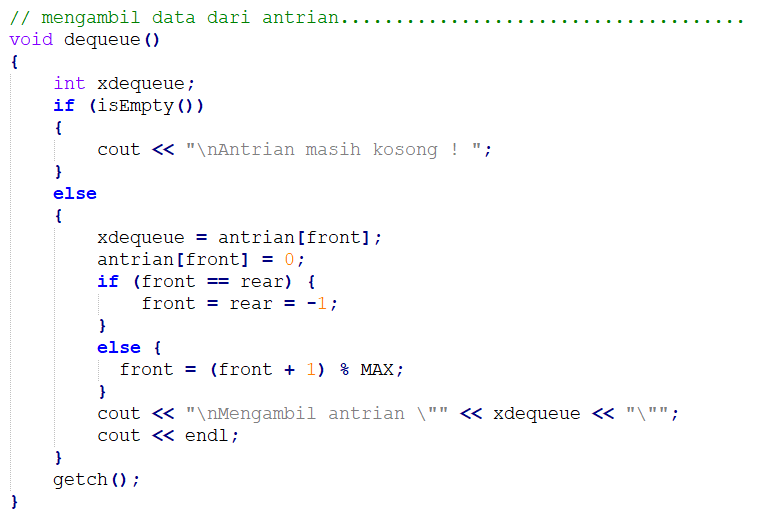
**Jumat**, 30 Oktober 2020

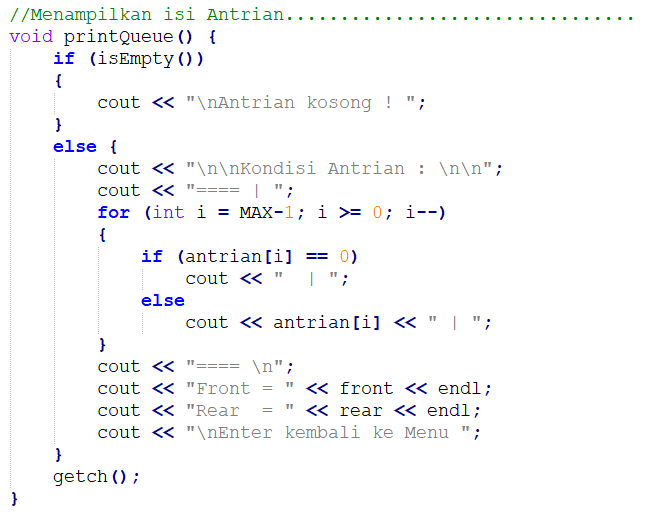
**Latihan 1 :**

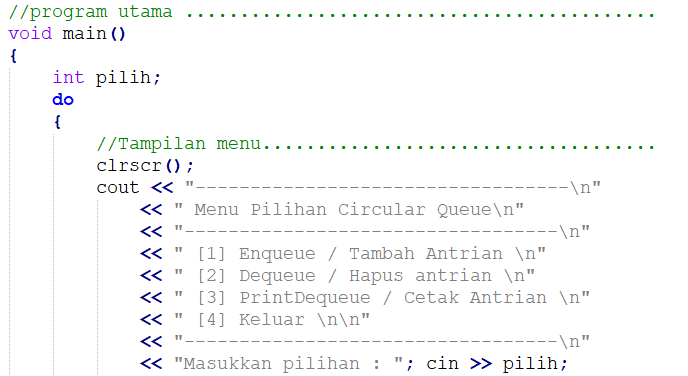
**Kode program :**

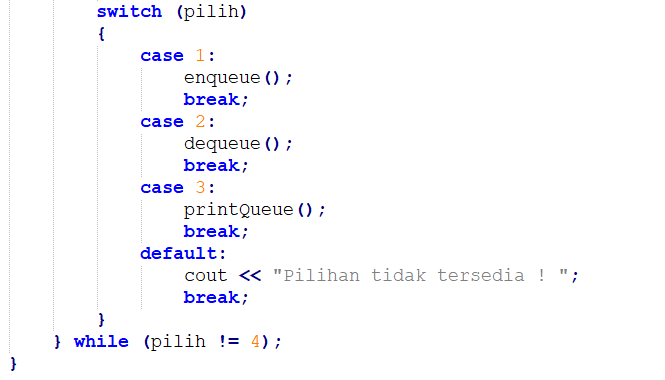
****

****

****

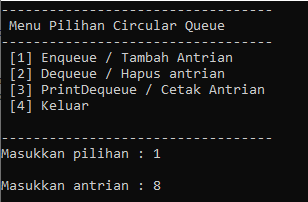
****

****

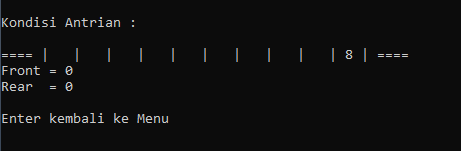
****

**Output program :**

* Enqueue / masukkan nilai 8 (pilih 1)

****

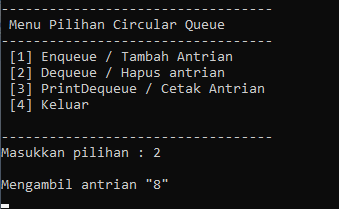
* Cetak Kondisi Antrian (pilih 3)

****

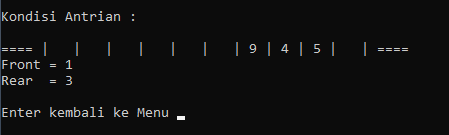
* Dimasukkan nilai 5, 4, dan 9 (pilih 1)
* Cetak Kondisi Antrian (pilih 3)

****

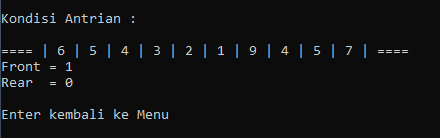
* Dequeue/hapus data Antrian (pilih 2)

****

* Cetak Kondisi Antrian (pilih 3)



* Dimasukkan nilai 1, 2, 3, 4, 5, 6, dan 7 (pilih 1)
* Cetak Kondisi Antrian (pilih 3)

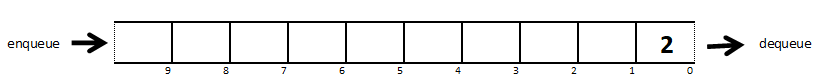
****

* Dequeue/hapus data Antrian (pilih 2)
* Cetak Kondisi Antrian (pilih 3)

****

**Tugas :**

Gambarkan kodisi antrian jika dilakukan proses berikut ini :

1. Enqueue 2
2. Enqueue 8, 3, 6, dan 9
3. dequeue
4. Enqueue 7, 5, 2, dan 4
5. Dequeue
6. Dequeue
7. Dequeue
8. Enqueue 1, 3, 6, dan 9
9. dequeue
10. dequeue

DIKUMPULKAN :

Nama File : **A12P99AlproXXXXXNama** (99 nomor tugas, XXXXX : 5 digit NIM terakhir)

Format file : PDF

Dikirim ke : [suharnawi@dsn.dinus.ac.id](mailto:suharnawi@dsn.dinus.ac.id)

Dikumpulkan paling lambat 7 November 2020

Source code program

#include <iostream.h>

#include <conio.h>

#include <stdio.h>

#define MAX 10

int antrian[MAX];

int front = -1;

int rear = -1;

//cek apakah antrian penuh.........................................

bool isFull()

{

if ((front == rear + 1) || (front == 0 && rear == MAX - 1)){

return true;

}

else

return false;

}

//cek apakah antrian kosong........................................

bool isEmpty()

{

return front == -1;

}

//manambahkan data ke antrian......................................

void enqueue() {

if (isFull())

{

cout << "\nAntrian penuh ! ";

getch();

}

else

{

int xantrian;

cout << "\nMasukkan antrian : ";cin >> xantrian;

if (front == -1)

front = 0;

rear = (rear + 1) % MAX;

antrian[rear] = xantrian;

}

}

// mengambil data dari antrian.....................................

void dequeue()

{

int xdequeue;

if (isEmpty())

{

cout << "\nAntrian masih kosong ! ";

}

else

{

xdequeue = antrian[front];

antrian[front] = 0;

if (front == rear) {

front = rear = -1;

}

else {

front = (front + 1) % MAX;

}

cout << "\nMengambil antrian \"" << xdequeue << "\"";

cout << endl;

}

getch();

}

//Menampilkan Queue

void printQueue() {

if (isEmpty())

{

cout << "\nAntrian kosong ! ";

}

else {

cout << "\n\nKondisi Antrian : \n\n";

cout << "==== | ";

for (int i = MAX-1; i >= 0; i--)

{

if (antrian[i] == 0)

cout << " | ";

else

cout << antrian[i] << " | ";

}

cout << "==== \n";

cout << "Front = " << front << endl;

cout << "Rear = " << rear << endl;

cout << "\nEnter kembali ke Menu ";

}

getch();

}

//program utama ...........................................

void main()

{

int pilih;

do

{

//Tampilan menu....................................

clrscr();

cout << "----------------------------------\n"

<< " Menu Pilihan Circular Queue\n"

<< "----------------------------------\n"

<< " [1] Enqueue / Tambah Antrian \n"

<< " [2] Dequeue / Hapus antrian \n"

<< " [3] PrintDequeue / Cetak Antrian \n"

<< " [4] Keluar \n\n"

<< "----------------------------------\n"

<< "Masukkan pilihan : "; cin >> pilih;

switch (pilih)

{

case 1:

enqueue();

break;

case 2:

dequeue();

break;

case 3:

printQueue();

break;

default:

cout << "Pilihan tidak tersedia ! ";

break;

}

} while (pilih != 4);

}