**QUEUE CIRCLE**

#include<stdio.h>

#include<conio.h>

//HANG DOI CAI DAT BANG MANG VONG

#define MaxLength 10

typedef int ElementType;

typedef struct

{

ElementType Elements[MaxLength];

int Front, Rear;

} Queue;

// TAO RONG

void MakeNull\_Queue(Queue \*Q){

Q->Front=-1;

Q->Rear=-1;

}

// KIEM TRA RONG

int Empty\_Queue(Queue Q){

return Q.Front==-1;

}

// KIEM TRA DAY

int Full\_Queue(Queue Q){

return (Q.Rear-Q.Front+1) % MaxLength==0;

}

// THEM PHAN TU MOI VAO HANG

void EnQueue(ElementType X, Queue \*Q){

if (!Full\_Queue(\*Q))

{

if (Empty\_Queue(\*Q)) Q->Front=0;

Q->Rear=(Q->Rear+1) % MaxLength;

Q->Elements[Q->Rear]=X;

}

else printf("Loi: Hang day!");

}

// XOA PHAN TU RA KHOI HANG - CAP NHAT LAI BIEN FRONT

void DeQueue(Queue \*Q){

if (!Empty\_Queue(\*Q))

{

if (Q->Front==Q->Rear) MakeNull\_Queue(Q);

else Q->Front=(Q->Front+1) % MaxLength;

}

else printf("Loi: Hang rong!");

}

// LAY NOI DUNG PHAN TU DAU HANG

ElementType Front(Queue Q){

return Q.Elements[Q.Front];

}

// TAO DANH SACH HANG DOI

void ReadQueue(Queue \*Q){

MakeNull\_Queue(Q);

ElementType X; int i=1;

do

{

printf("Phan tu thu %d: ",i); scanf("%d", &X);

if ((X!=-1) && (!Full\_Queue(\*Q)))

{ EnQueue(X,Q);

i++;

};

} while (X!=-1);

}

// IN TOAN BO HANG DOI RA MAN HINH

void PrintQueue(Queue \*Q){

while (!Empty\_Queue(\*Q))

{

printf("%d ", Front(\*Q));

DeQueue(Q);

}

printf("\n");

}

int main(){

Queue Q;

printf("Nhap du lieu vao hang doi KET THUC NHAN -1: \n ");

ReadQueue(&Q);

printf("\n Xoa phan tu dau ra khoi hang ");

DeQueue(&Q);

printf("\n Them phan tu co gia tri 10 vao hang: ");

EnQueue(10,&Q);

printf("\n Hang hien tai la: ");

PrintQueue(&Q);

if (Empty\_Queue(Q))

printf("\n\nHang con lai la hang rong ");

else

printf("\n\nHang con lai la khac rong !!!! ");

return 0;

}