// implementation of circular queue using linked list //

ALGORITHM:

Enqueue(data)

1. Create a struct node type node
2. Insert the given data in the new node data section and NULL in address
3. If queue is empty then initialize front and rear from new node
4. Queue is not empty then initialize rear next and rear from new node
5. New node next initialize from front

Dequeue()

1.check if queue is empty or not

2.if queue is empty then dequeue is not possible

3. else initialize temp from front

4. if front is equal to the rear then initialize front and rear from NULL

5. print data of temp and free temp memory

6. if there is more than one node in queue then make front next to front then initialize rear next from front

7. print temp and free temp

Display ()

1. Check if there is some data in the queue or not
2. If the queue is empty print “no data in the queue “
3. Else define a node pointer and initialize it with front
4. Print data of node pointer until the next of node pointer become NULL

// program to implementation circular queue in linked list //

#include<stdio.h>

#define max 6

Int queue[max];// array declaration

Int front =-1;

Int rear=-1;

//function to insert element in circular queue/

Void enqueue(int element)

{

If(front==-1&&rear==-1) // condition to check queue is empty //

{

Front=0;

Rear =0;

Queue[rear]=element;

}

Else if(( rear+1)%max=front) // condition checks queue is full//

{

Printf(“queue is overflow”);

}

Else

{

Rear=(rear+1)%max;

Queue[rear]=element;

}

}

Int dequeue()

{

If(front==-1)&&(rear==-1)

{

Print f(“\n queue is underflow”);

}

Else if(front==rear)

{

Print f(“\n the dequeue element is %d” ,queue[front]);

Front=-1;

Rear=-1;

}

Else

{

Print f(“\n the dequeue element is %d “, queue[front]);

Front=(front+1)%max;

}

}

Void display()

{

Int i=front;

If(front==-1&&rear==-1)

{

Printf(“\n queue is empty “);

}

Else

{

Printf(“\n element in a queue are :”);

While(i<=rear)

{

Printf(“%d”,queue[i]);

I=(i+1)%max;

}

}

}

Int main()

{

Int choice=1,x; //variable declaration//

While(choice<4&&choice!=0)// while loop

{

Printf(“\n press 1:insert an element”);

Printf(“\n press 2 : delete an element “);

Printf(“\ press 3: display an element “);

Printf(“\n enter the choice “);

Scanf(“%d”,&choice);

Switch(choice)

{

Case 1:

Printf(“ enter the element which is to be in inserted “);

Sacnf(“%d”,&x);

Enqueue(x);

Break:

Case 2:

Dequeue();

Break;

Case 3 :

Display();

}

}

Return 0;

}

OUT PUT



GIT HUB :