Dqueue

#include<iostream>

#define max 20

using namespace std;

class Queue{

public:

char queue[max],a,x;

int front,rear;

Queue(){

front = -1;

rear = -1;

a=0;

x=0;

}

bool is\_empty();

bool is\_full();

void insert\_Atfront(char a);

void insert\_Atrear(char x);

void del\_Atfront();

void del\_Atrear();

void display\_que();

};

bool Queue::is\_empty(){

if(front==-1&& rear==-1){

cout<<"queue is empty";

return true;

}

else{

cout<<"queue is not empty";

return false;

}

}

bool Queue::is\_full(){

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

cout<<"queue is full";

return true;

}

else{

cout<<"queue is not full";

return false;

}

return 0;

}

void Queue::insert\_Atfront(char a){

if(is\_full()){

cout<<"queue is full";

}

else if(front==-1){

front=rear=0;

cout<<"queue is empty";

}

else if(front==0){

front=max-1;

cout<<"front is at first position";

}

else{

front=front-1;

queue[front]=a;

cout<<"front is not at first position";

}

}

void Queue::insert\_Atrear(char x){

if(is\_full()){

cout<<"queue is full";

}

else if(front==-1){

front=rear=0;

cout<<"queue is empty";

}

else if(rear==max-1){

rear=0;

cout<<"rear is at last position";

}

else{

rear=rear+1;

queue[rear]=x;

cout<<"rear is not at end";

}

}

void Queue::del\_Atfront(){

if(is\_empty()){

cout<<"queue is empty";

cout<<"only one element";

}

else if(front==max-1){

front=0;

cout<<"queue is at end";

}

else{

front=front+1;

cout<<"front is not at end";

}

}

void Queue::del\_Atrear(){

if(is\_empty()){

cout<<"queue is empty";

cout<<"only one element is";

}

else if(rear==0){

rear=max-1;

cout<<"rear is at start";

}

else{

rear=rear+1;

cout<<"rear is not at start";

}

}

void Queue::display\_que(){

int i;

if(is\_empty()){

cout<<"queue is not empty";

}

for(i=(front+2);i<=rear;i++){

cout<<"Displaying the order";

}

}

int main()

{

Queue Q;

int ch;

Do{

cout<<”-----Menu—--------”;

cout<<"Enter 1 for Insert at Rear"<<endl;

cout<<"Enter 2 for Delete at Rear"<<endl;

cout<<"Enter 3 for Insert at Front"<<endl;

cout<<"Enter 4 for Delete at front"<<endl;

cout<<"Enter 5 for Display Queue"<<endl;

cout<<"Enter the valid choice"<<endl;

cin>>ch;

switch(ch){

case 1:

char A[7];

cin>>A;

Q.insert\_Atfront(A); // add orders function call

break;

case 2:

Q.insert\_Atrear(A); // delete order function call

break;

case 3:

Q.del\_Atfront();// getfront function call

break;

Q.del\_Atrear();

break;

Q.display\_que();

break;

}

ch++;

}

while(ch<=3);

cout<<"thank you";

return 0;}