**DSA LAB 5**

Arjun Mehta K036

B. Tech Cybersecurity

Code:

#include <iostream>

#include <limits>

#define MAX 5

using namespace std;

class Queue {

private:

int que[MAX];

int r,f;

public:

Queue():r(-1),f(-1){}

bool isEmpty(){

return (r==-1&&f==-1);

}

bool isFull(){

return ((r+1)%MAX==f);

}

int size(){

if (isEmpty()){

return 0;

} else if(r>=f){

return r-f+1;

} else{

return (MAX-f)+r+1;

}

}

int front(){

if(isEmpty()){

cout<<"Queue is empty"<<endl;

return -1;

} else{

return que[f];

}

}

void enqueue(int num){

if (isFull()){

cout<<"Queue is full"<<endl;

} else{

if(isEmpty()){

r=f=0;

} else{

r=(r+1)%MAX;

}

que[r]=num;

}

}

int dequeue(){

if (isEmpty()){

cout<<"Queue is empty"<<endl;

return -1;

} else{

int temp=que[f];

if(f==r){

f=r=-1;

} else{

f=(f+1)%MAX;

}

return temp;

}

}

void display(){

if (isEmpty()){

cout<<"Queue is empty"<<endl;

return;

}

int i=f;

do {

cout<<que[i]<<" ";

i=(i+1)%MAX;

} while(i!=(r+1)%MAX);

cout<<endl;

}

};

int main(){

Queue q;

int choice, value;

while(true) {

cout<<"\nCircular Queue Operations\n";

cout<<"1)Check if Empty\n";

cout<<"2)Size\n";

cout<<"3)Front\n";

cout<<"4)Display\n";

cout<<"5)Enqueue\n";

cout<<"6)Dequeue\n";

cout<<"7) Exit\n";

cout<<"Enter your choice now please= ";

cin>>choice;

switch(choice){

case 1:

cout<<(q.isEmpty()?"Queue is empty":"Queue is not empty")<<endl;

break;

case 2:

cout<<"Queue size="<<q.size()<<endl;

break;

case 3:

value = q.front();

if (value != -1) {

cout<<"Front value="<<value<<endl;

}

break;

case 4:

cout<<"Queue elements=";

q.display();

break;

case 5:

cout<<"Enter value to enqueu=";

cin>>value;

q.enqueue(value);

break;

case 6:

value=q.dequeue();

if (value!=-1){

cout<<"Dequeued value="<<value<<endl;

}

break;

case 7:

cout<<"Exiting program\n";

return 0;

default:

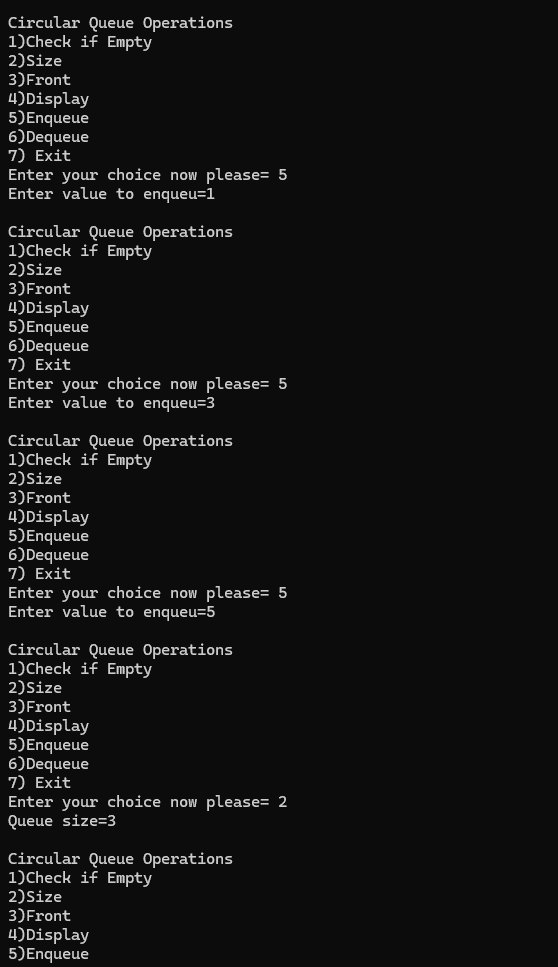
cout<<"Invalid choise.Please try again latee.\n";

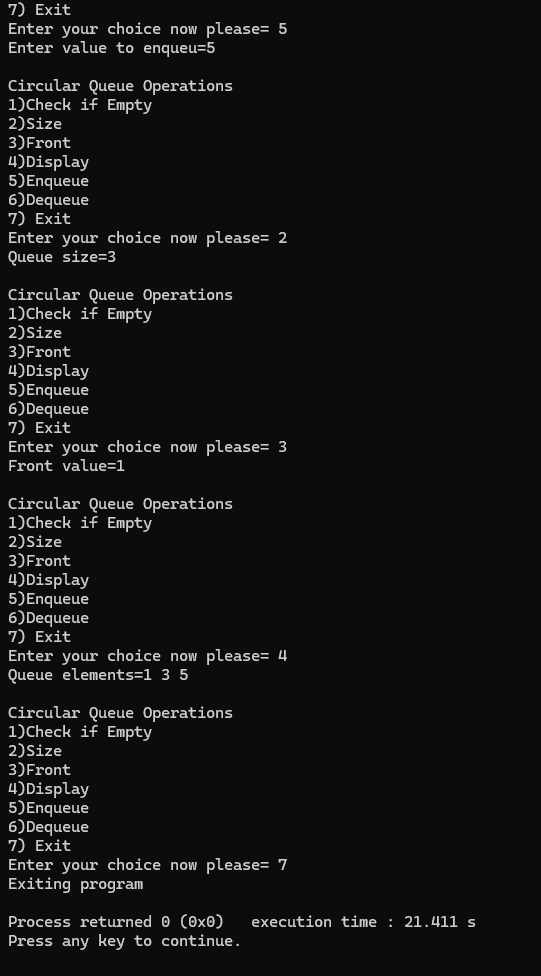
}

}

return 0;

}





Theory and Algorithms: -