Dsa Lab Final

Name Gotam kumar

Roll number : Cs211011

Code

class Execution{

int front;

int rear;

int size;

int[] arr;

int count;

public Execution(int size)

{

front=0;

rear=-1;

this.size=size;

arr=new int[size];

count = 0;

}

public void Enqueue(int id)

{

if(count == size){

System.out.println("Exection is full");

}

else{

rear=(rear+1)%size;

arr[rear]=id;

count++;

}

}

public int Dequeue(){

if(count==0)

{

System.out.println("Exection is Empty");

return null;

}

else

{

int temp=arr[front];

count--;

front=(front+1)%size;

return temp;

}

}

public int Peek(){

if(count==0){

System.out.println("Execution is Empty");

return null;

}

else{

return arr[front];

}

}

public void show(){

while(count!=0){

int temp=arr[front];

count--;

front=(front+1)%size;

System.out.println(" "+temp);

}

}

class Ready{

int front;

int rear;

int size;

int[] arr;

public Ready(int size)

{

front=0;

rear=-1;

this.size=size;

arr=new int[size];

}

public void Enqueue(int id)

{

if(count == size){

System.out.println("Ready is Full");

}

else{

rear=(rear+1)%size;

arr[rear]=id;

count++;

}

}

public int Dequeue(){

if(count==0)

{

System.out.println("ready is empty");

return null;

}

else

{

int temp=arr[front];

count--;

front=(front+1)%size;

return temp;

}

}

public int Peek(){

if(count == 0){

System.out.println("Ready is empty");

return null;

}

else{

return arr[front];

}

}

public void show(){

while(count!=0){

int temp=arr[front];

count--;

front=(front+1)%size;

System.out.println(" "+temp);

}

}

}

class node

{

int data;

node next;

public node(int data)

{

this.data=data;

next=null;

}

}

class waitingQ{

node front;

node rear;

public waitingQ()

{

rear=null;

front=null;

}

public void Enqueue(int id)

{

node n=new node(id);

if(rear == null)

{

front=n;

rear=n;

}

else

{

rear.next=n;

rear=n;

}

}

public int Dequeue()

{

if(front==null)

{

System.out.println("ready is empty");

return null;

}

else

{

int temp=front.id;

front=front.next;

return temp;

}

}

public int Peek()

{

if(front == null)

{

System.out.println("waiting q is empty");

return ;

}

else

{

return front.id;

}

}

}

class Manager {

String name;

String password;

public Manager(String name, String password) {

this.name = name;

this.password = password;

}

}

public class Main

{

Execution ex = new Execution(3);

Ready rd = new Ready(5);

waitingQ w = new waitingQ();

public void ADDShow()

{

for(int i =1; i<=10; i++)

{

System.out.println("plz enter your Id");

Scanner sc = new Scanner(System.in);

int id = new sc.nextInt();

if(ex.count < 3){

ex.Enqueue(id);

System.out.println("your are in Exection Queue");

ex.show();

}

else if (rd.count < 5){

rd.Enqueue(id);

System.out.println(your are in Ready queue);

rd.show();

}

else{

w.Enqueue(id);

System.out.println("your are in waiting");

w.show();

}

}

}

public void adding()

{

while (w.Peek() != null)

{

if(rd.count < 5)

{

int id = w.Dequeue();

rd.Enqueue(id);

}

else if(ex.count < 3){

int id = w.Dequeue();

ex.Enqueue(id);

}

}

}

public static void main(String[] args) {

Main m = new Main();

m.ADDShow();

}

}