JAVA PROGRAM:

**Question**- Prog lang: Java

In this problem, there are types of events: ENTER (a student enters the queue) or SERVED.

A unique token is assigned to any student entering the queue.

The queue serves the students based on the following criteria: The student having the highest Cumulative Grade Point Average (CGPA) is served first.

Any students having the same CGPA will be served by name in ascending case-sensitive alphabetical order

. Any students having the same CGPA and name will be served in ascending token order.

Given a sequence of events, print the names of students who are yet to be served(based on above criteria).

If the queue is empty, print EMPTY.

**Input Format**

The first line of input contains an integer, denoting the total number of events. Each of the subsequent lines will be of the following two forms:

ENTER name CGPA token - The student to be inserted into the priority queue.

SERVED - The highest priority student in the queue was served.

Constraints where where each token i is a unique integer.

**Output Format**

Print the names (based on the criteria) of the students who are not served at all after executing all events; if every student in the queue was served, then print EMPTY.

**Sample Input**

12

ENTER John 3.75 50

ENTER Mark 3.8 24

ENTER Shafaet 3.7 35

SERVED

SERVED

ENTER Samiha 3.85 36

SERVED

ENTER Ashley 3.9 42

ENTER Maria 3.6 46

ENTER Anik 3.95 49

ENTER Dan 3.95 50

SERVED

**Sample Output :**

Dan

Ashley

Shafaet

Maria

// PROGRAM STARTS FROM HERE

import java.util.\*;

class Data

{

String name;

double CGPA;

int token;

public Data(String name,float CGPA,int token) //constructor

{

this.name=name;

this.CGPA=CGPA;

this.token=token;

}

public String getname()

{

return name;

}

public double getCgpa()

{

return CGPA;

}

public int getToken()

{

return token;

}

}

public class Demo {

//MAIN FUNCTION STARTS HERE

public static void main(String args[])

{

int a=0;

Scanner Sc=new Scanner(System.in);

int n=Integer.parseInt(Sc.nextLine());

PriorityQueue<Data> Student = new PriorityQueue(Comparator.comparing(Data::getCgpa).reversed().thenComparing(Data::getname).thenComparing(Data::getToken));

// TAKING INPUT FROM USER

while(n-- >0)

{

String event=Sc.nextLine();

String eventarr[]=event.split(" ");

if(eventarr[0].equals("ENTER"))

{

String name=eventarr[1];

float CGPA=Float.parseFloat(eventarr[2]);

int token=Integer.parseInt(eventarr[3]);

Data d=new Data(name,CGPA,token);

Student.add(d);

}

else if(eventarr[0].equals("SERVED"))

{

Data Head=Student.poll();

}

}

// OUTPUT STARTS FROM HERE

Data Head=Student.poll();

if(Head==null)

{

System.out.println("EMPTY");

}

else

{

while(Head!=null)

{

System.out.println(Head.name);

Head=Student.poll();

}

}

}

}