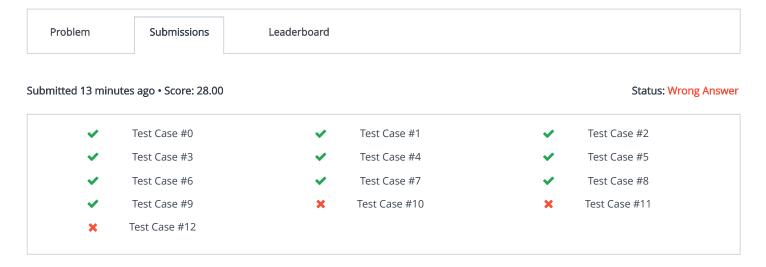
Q Search

adrian_santacruz 🗸

All Contests > HW4_FIRST_CONTEST > hw4_e4) Priority Queue

hw4_e4) Priority Queue



Submitted Code

```
Language: Python 3
                                                                                              P Open in editor
 1 def createHeap(arr, linkedarr):
       for i in range(1,len(arr)):
 2
3
           child = i
           parent = (i-1)//2
 4
 5
           while child>0 and arr[parent]<arr[child]:</pre>
               arr[parent],arr[child] = arr[child],arr[parent]
 7
               linkedarr[parent],linkedarr[child] = linkedarr[child],linkedarr[parent]
               child = parent
8
9
               parent= (child-1)//2
10
11 def PriorityQueue(people, grades, top):
12
       for i in range(top):
13
           biggest_index = 0
14
           createHeap(grades, people)
15
           for i in range(1, len(people)):
               if grades[i] > grades[biggest_index]:
16
                   biggest_index = i
17
18
19
           print(people[biggest_index], grades[biggest_index])
20
           print(*grades)
21
           p = people.pop(biggest_index)
22
           g = grades.pop(biggest_index)
23
24 numbers = input()
25 numbers = list(map(int,numbers.split()))
26 total_people = numbers[0]
27 output_people = numbers[1]
28
29 people_and_grades = []
30
```

```
31 for i in range(total_people):
       people_and_grades.append(input())
32
33
34 people = []
35 grades = []
36
37 L = []
38
39 def itself(x):
40
       return x
41
42 for element in people_and_grades:
       person_and_grade = list(map(itself, element.split()))
43
44
       L.append(person_and_grade)
45
       person = person_and_grade[0]
46
       grade = person_and_grade[1]
47
      people.append(person)
48
       grades.append(grade)
49
50 output = PriorityQueue(people, grades, output_people)
51
```

Interview Prep | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy |