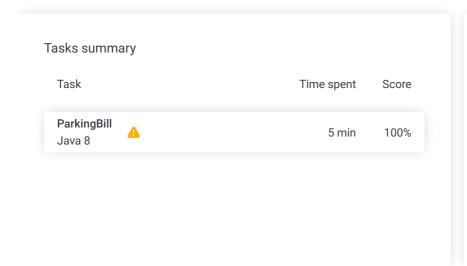
Codility_

CodeCheck Report: trainingMN3P82-M94

Test Name:

Check out Codility training tasks

Summary Timeline 👜 Al Assistant Transcript





Tasks Details

1. ParkingBill

lementary

Given two strings representing times of entry and exit from a car parking lot, find the cost of the ticket according to the given billing rules.

Task Score Correctness Performance
100% 100% Not assessed

Task description

You parked your car in a parking lot and want to compute the total cost of the ticket. The billing rules are as follows:

- The entrance fee of the car parking lot is 2;
- The first full or partial hour costs 3;
- Each successive full or partial hour (after the first) costs 4.

You entered the car parking lot at time E and left at time L. In this task, times are represented as strings in the format "HH:MM" (where "HH" is a two-digit number between 0 and 23, which stands for hours, and "MM" is a two-digit number between 0 and 59, which stands for minutes).

Write a function:

class Solution { public int solution(String E,
String L); }

that, given strings E and L specifying points in time in the format "HH:MM", returns the total cost of the parking bill from your entry

Solution

Effective time used:

Programming language used: Java 8

Total time used: 5 minutes

5 minutes

Notes: not defined yet

08:29:19 08:34:03

Code: 08:34:03 UTC, java, show code in pop-up final, score: 100

at time E to your exit at time L. You can assume that E describes a time before L on the same day.

For example, given "10:00" and "13:21" your function should return 17, because the entrance fee equals 2, the first hour costs 3 and there are two more full hours and part of a further hour, so the total cost is 2 + 3 + (3 * 4) = 17. Given "09:42" and "11:42" your function should return 9, because the entrance fee equals 2, the first hour costs 3 and the second hour costs 4, so the total cost is 2 + 3 + 4 = 9.

Assume that:

- strings E and L follow the format "HH: MM" strictly;
- string E describes a time before L on the same day.

In your solution, focus on **correctness**. The performance of your solution will not be the focus of the assessment.

Copyright 2009–2024 by Codility Limited. All Rights Reserved. Unauthorized copying, publication or disclosure prohibited.

Test results - Codility

```
// you can also use imports, for example:
     // import java.util.*;
3
     import java.text.ParseException;
     import java.text.SimpleDateFormat;
4
     import java.util.Date;
6
7
     // you can write to stdout for debugging purposes,
     // System.out.println("this is a debug message");
8
9
10
     class Solution {
11
         public int solution(String E, String L) {
12
             // Implement your solution here
13
             int totalCost = 0;
14
                     SimpleDateFormat format = new Simp
15
16
                     long difference=0;
17
                     try {
                             Date date1 = format.parse(
18
                             Date date2 = format.parse(
19
20
                             difference = date2.getTime
                     } catch (ParseException e) {
21
22
                              // TODO Auto-generated cat
23
                             e.printStackTrace();
24
25
26
             int diffMinutes = (int)(long)(difference /
27
                     int diffHours = (int)(long)differe
28
                     if(diffMinutes>0) {
29
30
                             diffHours+=1;
31
32
                     int totalBillableHours=diffHours;
33
                     totalCost=2+3+((totalBillableHours
34
35
                     return totalCost;
36
37
     }
```

Analysis summary

The solution obtained perfect score.

Analysis

expand all	Example test	S	
example1 first example test	t	✓	OK
example2 second example	test	✓	ОК
expand all	Correctness tes	sts	3
under_ten_min very short parking always 5	nutes g times, answer is	√	OK
random_undeshort parking tim	er_hour es, answer is always	√	OK
equal_hours_s parking for short complete hours	small time, always with	√	OK
random randomly general	ted test cases	✓	ОК
► mixed medium and sho	rt intervals	✓	ОК

Test results - Codility

	ual_hours_big g parking time, for complete hours	✓ OK	
•	random_big randomly generated parking time at least 20 hours	✓ OK es, for	
•	maximum_result test cases giving maximum resu almost maximum results	✓ OK Its or	