

paiza - C076 : 문제

9시부터 17시까지 : 시급 X 엔 (통상 시급)

17시부터 22시까지 : 시급 Y 엔 (야간 시급)

· 그 이외의 시간 : 시급 Z 엔 (심야 시급)

당신 N 일 사이 출근 시간과 퇴근 시간을 1 시간 단위로 주어집니다.

N 일 총으로 얻을 수 있는 금액을 계산합니다.

입력 예 1의 경우 4 일 동안 시급 X 엔, Y 엔, Z 원 형이 되는 시간대를 두루 근무하며

총 금액은 $Z \times 9 + X \times 8 + Y \times 5 + Z \times 1 = 1500 \times 9 + 1000 \times 8 + 1300 \times 5 + 1500 \times 1 = 29500$ 엔입니다.

- 입력되는 값 입력은 다음 형식으로 제공됩니다.
X Y Z • 첫 번째 줄에는 보통 시급 X, 밤 시급 Y 심야 시급 Z가이 순서로 정수 공백으로 구분하여 부여됩니다.
N • 두 번째 줄에는 출근 일수 N은 정수로 주어집니다.
S_1 T_1 • 이어지는 N 행의 i 번째 (1 ≤ i ≤ N)는 i 일째 출근 시간 S_i과 퇴근 시간 T_i가이 순서로 정수 공백으로 구
S_2 T_2 분하여 부여됩니다.
... • 입력은 총 N + 2 행이 입력 값 마지막 줄의 끝에 줄 바꿈이 하나 들어갑니다.
S_N T_N • 각각의 값은 문자열로 표준 입력으로부터 전달됩니다

조건
모든 테스트 케이스에서 다음의 조건을 채우고 있습니다.

- $2.0 \leq X, Y, Z \leq 3,000$
- $1 \leq N \leq 100$
- $2.0 \leq S < T \leq 23$

입력 예 1 7
1000 1300 1500 8 19
4 9 20
0 9 10 21
9 17 11 22
17 22 0 23
22 23 20 22
 0 21

출력 예1 출력 예2
29500 130000

paiza - C076 : 채점 결과

제출 코드 결과 상세

테스트 번호	입력 케이스 번호	심판 결과	실행 시간
테스트 1	케이스 1 (기본 데이터)	 통과	0.08 초
테스트 2	케이스 1 (기본 데이터)	 통과	0.08 초
테스트 3	케이스 1 (기본 데이터)	 통과	0.10 초
테스트 4	케이스 1 (기본 데이터)	 통과	0.08 초
테스트 5	케이스 1 (기본 데이터)	 통과	0.09 초
테스트 6	케이스 1 (기본 데이터)	 통과	0.08 초
테스트 7	케이스 1 (기본 데이터)	 통과	0.08 초
테스트 8	케이스 1 (기본 데이터)	 통과	0.09 초
테스트 9	케이스 1 (기본 데이터)	 통과	0.08 초
테스트 10	케이스 1 (기본 데이터)	 통과	0.08 초

paiza - C076 : 코드

```
*C076.java
1 package paiza;
2
3 import java.util.*;
4
5 public class C076 {
6
7     public static void main(String[] args) {
8         Scanner sc = new Scanner(System.in);
9         int dayWage = Integer.parseInt(sc.next());
10        int eveWage = Integer.parseInt(sc.next());
11        int nightWage = Integer.parseInt(sc.next());
12        int workingDay = Integer.parseInt(sc.next());
13        int totalWage = 0;
14        /* 9~17: day
15         * 17~22: eve
16         * else: night */
17        for(int i = 0; i < workingDay; i++) {
18
19            int openingHour = Integer.parseInt(sc.next());
20            int closingHour = Integer.parseInt(sc.next());
21            // openingHour > 9 and closingHour < 17 -> dayWage 적용
22            // openingHour > 9 and (closingHour > 17 and closingHour < 22)
23            // -> dayWage + eve 적용
24            // else -> dayWage + eve + night 적용
25
26            if(openingHour >= 9 && openingHour < 17) {
27                if(closingHour > 9 && closingHour <= 17) {
28                    totalWage += dayWage * (closingHour - openingHour);
29                } else if(closingHour > 17 && closingHour <= 22) {
30                    totalWage += dayWage * (17 - openingHour)
31                        + eveWage * (closingHour - 17);
32                } else if(closingHour > 22 && closingHour <= 24) {
33                    totalWage += dayWage * (17 - openingHour)
34                        + eveWage * (22 - 17)
35                        + nightWage * (closingHour - 22);
36                }
37            }
38        }
39    }
40}
```

paiza - C076 : 코드

```
37         } else if(openingHour >= 17 && openingHour < 22) {
38             if(closingHour > 17 && closingHour <= 22) {
39                 totalWage += eveWage * (closingHour - openingHour);
40             } else if(closingHour > 22 && closingHour <= 24) {
41                 totalWage += eveWage * (22 - openingHour)
42                     + nightWage *(closingHour - 22);
43             }
44         } else if(openingHour >= 22) {
45             totalWage += nightWage *(closingHour - openingHour);
46         } else { // openingHour < 9
47             if(closingHour <= 9) {
48                 totalWage += nightWage * (closingHour - openingHour);
49             } else if(closingHour >= 9 && closingHour < 17) {
50                 totalWage += nightWage * (9 - openingHour)
51                     + dayWage * (closingHour - 9);
52             } else if(closingHour >= 17 && closingHour <22) {
53                 totalWage += nightWage * (9 - openingHour)
54                     + dayWage * (17 - 9)
55                     + eveWage * (closingHour - 17);
56             } else {
57                 totalWage += nightWage * (9 - openingHour)
58                     + dayWage * (17 - 9)
59                     + eveWage * (22 - 17)
60                     + nightWage * (closingHour - 22);
61             }
62         }
63     }
64 }
65 System.out.println(totalWage);
66 }
67
68 }
```