



# Drive

by [wanbo](#)

Problem

Submissions

Leaderboard

Discussions

HackerRank is starting a bus service in [MountainView, California](#). The bus starts at time  $T = 0$  at  $station_1$  and goes through  $station_2$ ,  $station_3$ ,  $station_4$  in that order and reaches the headquarters located at  $station_n$ . At every station, the bus waits for various commuters to arrive before it departs to the next station. Ignoring the acceleration, the bus moves at 1 meter / second. i.e., if  $station_i$  and  $station_j$  are 1000 meters apart, the bus takes 1000 seconds to travel from  $station_i$  to  $station_j$ .

The bus is equipped with  $K$  units of Nitro ( $N_2O$ ). If going from  $station_i$  to  $station_j$  takes  $x$  seconds, then using  $t$  units of nitro can decrease the time taken to  $\max(x-t, 0)$  seconds where  $\max(a,b)$  denotes the greater of the two values between  $a$  &  $b$ . The Nitro can be used all at once or in multiples of 1 unit.

If the bus driver travels optimally, what is the minimum sum of travelling time for all commuters? The travelling time equals to the time he/she arrived at the destination minus the time he/she arrived the start station.

Please remember that the driver must take all passengers to their destination.

## Input Format

The first line contains 3 space separated integers  $n$ ,  $m$  and  $K$  which indicate the number of stations, total number of people who board the bus at various stations and the total units of Nitro ( $N_2O$ ) present in the bus.

The second line contains  $n-1$  space separated integers where the  $i^{\text{th}}$  integer indicates the distance between  $station_{(i-1)}$  to  $station_i$ .

$m$  lines follow each containing 3 space separated integers. The  $i^{\text{th}}$  line contains  $t_i$ ,  $s_i$  and  $e_i$  in that order indicating the arrival time of the commuter at  $s_i$  at time  $t_i$  with his destination being  $e_i$ .

```
n m K
d1 d2 ... dn-1 // di: the distance between station_i to station_(i+1).
t1 s1 e1      // commuter 1 arrives at his boarding point at s1 and his destination is e1
t2 s2 e2
...
tm sm em
```

## Constraints

```
0 < n <= 100000
0 < m <= 100000
0 <= K <= 10000000
0 < d_i <= 100
0 <= t_i <= 10000000
1 <= s_i < e_i <= n
```

## Output Format

The minimal total travel time.

## Sample Input

```
3 3 2
1 4
1 1 3
```

```
2 1 2
5 2 3
```

### Sample Output

9

### Explanation

The bus waits for the 1<sup>st</sup> and the 2<sup>nd</sup> commuter to arrive at station<sub>1</sub> and travels to station<sub>2</sub> carrying 2 passengers. The travel time from station<sub>1</sub> to station<sub>2</sub> is 1 second. It then waits for the 3<sup>rd</sup> commuter to board the bus at time = 5, 2<sup>nd</sup> commuter deboards the bus. The 3<sup>rd</sup> commuter boards the bus at t = 5. The bus now uses 2 units of nitro, this reduces the commute time to travel to station<sub>3</sub> from 4 to 2.

Hence, the total time spent by each of the passengers on the bus is

- 1 (time spent waiting for commuter 2) + 1 (travel time from station<sub>1</sub> to station<sub>2</sub>) + 2 (time spent waiting for commuter 3) + 2 (travel time from station<sub>2</sub> to station<sub>3</sub>) = 6
- 1 (travel time from station<sub>1</sub> to station<sub>2</sub>)
- 2 (travel time from station<sub>2</sub> to station<sub>3</sub>)

$$6+1+2 = 9$$

hence the answer.

### Timelimits

Timelimits for this challenge can be seen [here](#)

f t in

Submissions:99

Max Score:90

Difficulty: Expert

Rate This Challenge:

☆☆☆☆☆

[More](#)

Current Buffer (saved locally, editable) 🔗 ↺

Java 7



```
1 import java.io.*;
2 import java.util.*;
3 import java.text.*;
4 import java.math.*;
5 import java.util.regex.*;
6
7 public class Solution {
8
9     public static void main(String[] args) {
10         /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
11     }
12 }
```

Line: 1 Col: 1

[Upload Code as File](#) ☐ Test against custom input

Run Code

Submit Code

