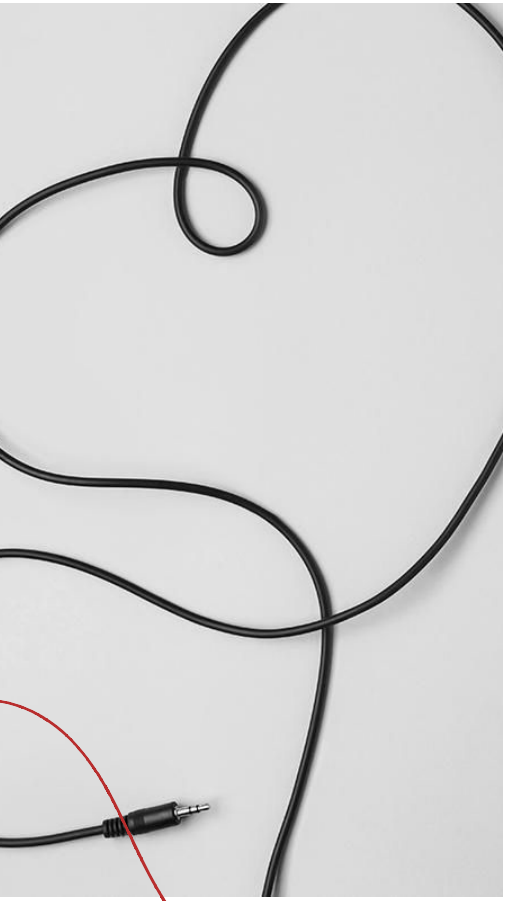

FROG JUMP

CSX 3009 Algorithm Design Term Project

Problem:

Educational DP Contest AtCoder A - Frog 1

From Dmoj.ca Point 7

Saw Zwe Wai Yan 6318013

TABLE OF CONTENTS

PROBLEM

There are N stones, numbered $1, 2, \ldots, N$. For each i $(1 \leq i \leq N)$, the height of Stone i is h_i.

There is a frog who is initially on Stone 1. He will repeat the following action some number of times to reach Stone N:

- If the frog is currently on Stone i, jump to Stone $i+1$ or Stone $i+2$. Here, a cost of $|h_i - h_j|$ is incurred, where j is the stone to land on.

Find the minimum possible total cost incurred before the frog reaches Stone N.

Constraints

- All values in input are integers.
- $2 \leq N \leq 10^5$
- $1 \leq h_i \leq 10^4$

Input Specification

The first line of input will contain an integer N.

The second line of input will contain N spaced integers, h_i, the height of stone i.

Output Specification

Output a single integer, the minimum possible total cost incurred.

ANALYSIS

The problem is to find the shortest distance the frog jump to reach to the last stone.

The cost to jump to the next stone is [previous cost + |(Stonei - Stonej)|]

The frog can jump i+1 or i+2 stones from from the current stone position.

The frog cannot go back in reverse.

CODE & OUTPUT

```python
N = int(input()) #Num of stones
stones = list(map(int, input().split())) #Stones

cost = [0] * N
cost[1] = abs(stones[1] - stones[0])

for i in range(2, N):
    cost[i] = min(cost[i-2] + abs(stones[i] - stones[i-2]), cost[i-1] + abs(stones[i] - stones[i-1]))
print(cost[N-1])
```

Resources: 0.737s, 20.77 MB
Maximum single-case runtime: 0.108s
Final score: 100/100 (7.0/7 points)

CODES AND OUTPUT

Sample Input 1

```
4
10 30 40 20
```

Copy

Sample Output 1

```
30
```

Copy

Sample Input 2

```
2
10 10
```

Copy

Sample Output 2

```
0
```

Copy

TEST CASE EXPLAINED

Test Case 1

N = 4 stones = [10 , 30 , 40 , 20] (0 1 2 3) | i = 3

cost = [0 , 20 , 0 , 0]

For Loop starts i = 2 till N

Compare min cost

10 → 30 → 40 20

i = 2

stones = [10 , 30 , 40 , 20] (0 1 2 3) cost = [0 , 20 , 0 , 0] (0 1 2 3)

cost [2] = min (cost [0] + abs [stones [2] - stones [0] , cost [1] + abs (stones [2] - stones [1]))

= min (0 + |40 - 0| , 20 + |40 - 30|)

= min (40 , 30)

= 30

∴ Cost = [0 , 20 , 30 , 0]

TEST CASE EXPLAINED

$i = 3$

stones = [10, 30, 40, 20] (0, 1, 2, 3) cost = [0, 20, 30, 0] (0, 1, 2, 3)

Compare min cost

10 30 40 20

$cost[2] = \min(cost[1] + abs(stones[3] - stones[1]), cost[2] + abs(stones[3] - stones[2]))$

$= \min(20 + |20 - 30|, 30 + |20 - 40|)$ (10, 20)

$= \min(30, 50)$

$= 30$

$\therefore cost = [0, 20, 30, 30]$

Final Output

Minimum cost = 30

REFERENCES

Dmoj.ca Problem:
https://dmoj.ca/problem/dpa

Problem Explanation:
https://www.youtube.com/watch?v=AlRvtF68B70

THANK YOU