Tight Arrays

locked

by kevinsogo

Editorial by kevinsogo

Suppose we stand on the number line, and at every second, we can either go left one step, go right one step, or stay at our current position. Also, suppose we record our position in an array every second. Then the resulting array is a *tight* array!

Conversely, we can associate any tight array to a certain walk on the number line. Now, we can rephrase the problem:

Suppose that you stand on position a on the number line, and you want to end up at c, and you need to pass by b along the way. What's the shortest time needed to accomplish that?

Now, we can answer this problem by noting that it takes $|a-b|$ seconds to go from a to b, and $|b-c|$ seconds to go from b to c (where $|x|$ is the absolute value of x). Therefore, the answer is $1+|a-b|+|b-c|$. Note that we need to add 1 since we need to count the starting position as well.

Absolute value

The formula "$1+|a-b|+|b-c|$" is easy to implement, assuming that we can compute absolute values. In most programming languages, the `abs` function is present as a built-in or a library function. But it's not that hard to implement your own as well. For example, here's a solution template in C++:

```
#include <iostream>
using namespace std;

int abs(int x) {
    // ...
}
int main() {
    int a, b, c;
    cin >> a >> b >> c;
    cout << 1 + abs(a - b) + abs(b - c) << endl;
}
```

Now, we need to implement the `abs` function. Here are a couple of ways. The simplest one is to use an `if-else` statement:

```
int abs(int x) {
    if (x > 0) {
        return x;
    } else {
        return -x;
    }
}
```

A shorter way to write this is with a *ternary* expression:

```
int abs(int x) {
    return x > 0 ? x : -x;
}
```

Statistics

Difficulty: Easy
Time Complexity: $O(1)$
Required Knowledge: Conditionals, absolute value
Publish Date: Apr 07 2017

 Set by kevinsogo

Problem Setter's code :

Python

```
a, b, c = map(int, raw_input().strip().split())
print 1 + abs(a - b) + abs(b - c)
```

 Tested by Shafaet

Problem Tester's code :

Ruby

```
#!/bin/ruby

a,b,c = gets.strip.split(' ')
a = a.to_i
b = b.to_i
c = c.to_i

if a>=1 and b>=1 and c>=1 and a<=100 and b<=100 and c<=100
    ans = (a-b).abs + (b-c).abs + 1
    print ans
end
```