The Hurdle Race

Editorial by AllisonP

Observation

We know the character can jump any hurdle that has height $h[i] \leq k$. To find the minimum number of beverages to clear all the hurdles, we simply need to determine the difference between the height they can already jump and the height they need to jump in order to complete the race. Note that if each $h[i] \leq k$, they can already clear all the hurdles and the answer is 0.

Solution

1. Find the maximum height of any hurdle, h_{max}.
2. Calculate the difference between h_{max} and k as $difference = h_{max} - k$.
   - If $difference > 0$, then the difference corresponds to the number of beverages he must drink and we print the value of $difference$ as our answer.
   - If $difference \leq 0$, then he is already capable of clearing all the hurdles and we print 0 as our answer.

Set by ma5termind

Problem Setter's code:

Cpp

```
#include <bits/stdc++.h>
using namespace std;


int main() {
        int n, k;
        cin >> n >> k;
    assert(n > 0 && n <= 100);
    assert(k > 0 && k <= 100);
        int mx = 0;
        while(n -- ){
                int x; cin >> x;
        assert(x > 0 && x <= 100);
                mx = max(mx, x);
        }
        cout << max(0, mx - k) << "\n";
        return 0;
}
```

Tested by jpierce88

Problem Tester's code:

javascript

```
function hurdleRace(k, height) {
    return Math.max(0, Math.max(...height) - k);
}
```

STATISTICS

Difficulty:	Easy
Time Complexity:	O(N)
Required Knowledge:	Adhoc, Implementation, Brute Force
Publish Date:	Jan 26 2017

Originally featured in HourRank 17

NEED HELP?

View discussions

View top submissions

Feedback

Was this editorial helpful?

Yes No