

Qualification Round 2011

A. Bot Trust

B. Magicka

C. Candy Splitting

D. GoroSort

Contest Analysis

Questions asked 3



Submissions

Bot Trust

10pt | Not attempted 10560/12572 users correct (84%) 10pt | Not attempted 10291/10514

users correct (98%)

Magicka

10pt Not attempted 8886/10218 users correct (87%)

15pt Not attempted 7176/8738 users correct (82%)

Candy Splitting

15pt Not attempted 6286/7416 users correct (85%)

GoroSort

correct (97%)

Top Scores	
SkidanovAlexander	100
tomconerly	100
kmod	100
watashi	100
RAD.	100
Anton.Lunyov	100
w01fe	100
jakubr	100
Weiqi	100
hos.lyric	100

Problem A. Bot Trust

This contest is open for practice. You can try every problem as many times as you like, though we won't keep track of which problems you solve. Read the <u>Quick-Start Guide</u> to get started.

Small input 10 points

Large input
10 points

Solve A-large

Solve A-small

Problem

Blue and Orange are friendly robots. An evil computer mastermind has locked them up in separate hallways to test them, and then possibly give them cake.

Each hallway contains 100 buttons labeled with the positive integers {1, 2, ..., 100}. Button k is always k meters from the start of the hallway, and the robots both begin at button 1. Over the period of one second, a robot can walk one meter in either direction, or it can press the button at its position once, or it can stay at its position and not press the button. To complete the test, the robots need to push a certain sequence of buttons in a certain order. Both robots know the full sequence in advance. How fast can they complete it?

For example, let's consider the following button sequence:

0 2, B 1, B 2, 0 4

Here, $0\ 2$ means button 2 in Orange's hallway, $B\ 1$ means button 1 in Blue's hallway, and so on. The robots can push this sequence of buttons in 6 seconds using the strategy shown below:

Time Orange	Blue
1 Move to button 2 2 Push button 2 3 Move to button 3 4 Move to button 4 5 Stay at button 4 6 Push button 4	P Stay at button 1 Stay at button 1 Push button 1 Move to button 2 Push button 2

Note that Blue has to wait until Orange has completely finished pushing 0 $\,$ 2 before it can start pushing B $\,$ 1.

Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow.

Each test case consists of a single line beginning with a positive integer \mathbf{N} , representing the number of buttons that need to be pressed. This is followed by \mathbf{N} terms of the form " \mathbf{R}_i \mathbf{P}_i " where \mathbf{R}_i is a robot color (always 'O' or 'B'), and \mathbf{P}_i is a button position.

Output

For each test case, output one line containing "Case #x: y", where x is the case number (starting from 1) and y is the minimum number of seconds required for the robots to push the given buttons, in order.

Limits

 $1 \le \mathbf{P}_i \le 100$ for all i.

Small dataset

 $1 \le \mathbf{T} \le 20.$ $1 \le \mathbf{N} \le 10.$

Large dataset

 $1 \le \mathbf{T} \le 100.$ $1 \le \mathbf{N} \le 100.$

Sample

Input	Output
3 4 0 2 B 1 B 2 0 4	Case #1: 6 Case #2: 100
3 0 5 0 8 B 100 2 B 2 B 1	Case #3: 4

 $\textbf{All problem statements, input data and contest analyses are licensed under the} \ \underline{\textbf{Creative Commons Attribution License}}.$

© 2008-2017 Google Google Home - Terms and Conditions - Privacy Policies and Principles

Powered by



Google Cloud Platform