

World Finals 2015

[A. Costly Binary Search](#)

[B. Campinatorics](#)

**C. Pretty Good Proportion**

[D. Taking Over The World](#)

[E. Merlin QA](#)

[F. Crane Truck](#)

[Contest Analysis](#)

[Questions asked](#)

Submissions

Costly Binary Search

8pt	Not attempted 20/25 users correct (80%)
19pt	Not attempted 16/17 users correct (94%)

Campinatorics

6pt	Not attempted 25/25 users correct (100%)
21pt	Not attempted 23/25 users correct (92%)

Pretty Good Proportion

5pt	Not attempted 26/26 users correct (100%)
22pt	Not attempted 10/18 users correct (56%)

Taking Over The World

7pt	Not attempted 20/21 users correct (95%)
29pt	Not attempted 3/4 users correct (75%)

Merlin QA

8pt	Not attempted 14/19 users correct (74%)
30pt	Not attempted 4/8 users correct (50%)

Crane Truck

8pt	Not attempted 2/3 users correct (67%)
37pt	Not attempted 0/1 users correct (0%)

Top Scores

Gennady.Korotkevich	155
rng..58	134
bmerry	104
tczajka	96
vepifanov	96
peter50216	96
tkociumaka	96
linguo	92
simonlindholm	77
pashka	76

Problem C. Pretty Good Proportion

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 [Quick-Start Guide](#) to get started.

Small input  
5 points

Solve C-small

Large input  
22 points

Solve C-large

Problem

I have a sequence of **N** binary digits. I am looking for a substring with just the right proportion of 0s and 1s, but it may not exist, so I will settle for something that's just pretty good.

Can you find a substring where the fraction of 1s is as close as possible to the given fraction **F**? Output the earliest possible index at which such a substring starts.

Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow. Each one starts with a line containing **N** and **F**. **F** will be a decimal fraction between 0 and 1 inclusive, with exactly 6 digits after the decimal point. The next line contains **N** digits, each being either 0 or 1.

Output

For each test case, output one line containing "Case #x: y", where x is the test case number (starting from 1) and y is the 0-based index of the start of the substring with the fraction of 1s that is as close as possible to **F**. If there are multiple possible answers, output the smallest correct value.

Limits

$1 \leq T \leq 100$ .  
 $0 \leq F \leq 1$   
**F** will have exactly 6 digits after the decimal point.

Small dataset

$1 \leq N \leq 1000$ .

Large dataset

$1 \leq N \leq 500,000$ .

Sample

Input	Output
5	Case #1: 5
12 0.666667	Case #2: 5
001001010111	Case #3: 5
11 0.400000	Case #4: 0
10000100011	Case #5: 6
9 0.000000	
111110111	
5 1.000000	
00000	
15 0.333333	
000000000011000	

In Case #1, there is no substring that has exactly a 1-proportion of exactly 666667/1000000. The closest we can get is 2/3. The input string has 5 substrings that achieve it -- 3 substrings of length 3 that start at indices 5, 7, and 8 (101, 101, and 011); as well as two substrings of length 6 that start at indices 5 and 6 (101011 and 010111). The smallest of these indices is 5.

---

All problem statements, input data and contest analyses are licensed under the [Creative Commons Attribution License](#).

© 2008-2017 Google [Google Home](#) - [Terms and Conditions](#) - [Privacy Policies and Principles](#)

Powered by



Google Cloud Platform