2014 Mind Sports Olympiad Computer Programming Competition

You can do any question in any order. In order to claim points, when you are ready to submit an answer to a particular question, call the examiner who will apply several test cases to your submission and score that problem (which cannot be resubmitted). You will have until 10pm to claim points and submit answers *[start was 7.30 – Julia]*, with the exception that the bonus points for Sudoku will be given after all participants have finished and submitted their solution.

**Optimise Joe’s Schedule (8 points)**

Joe goes to school in a very tall building. He attends N classes during the day, numbered chronologically from 1 to N. Class I is located on floor a(i). Joe’s schedule allows him to switch the order of two consecutive classes. Since Joe does not like to walk upstairs, he wants to choose the schedule that requires him to walk up the least number of stairs throughout the day. (Each pair of consecutive floors is separated by the same number of stairs). Help Joe determine which classes to switch to minimize the number of stairs he must walk UP during the school day, assuming he starts counting at the beginning of his first class.

Input: The first line of the input contains a single integer N. The second line of the input contains N space-separated integers a(1), a(2), …, a(N).

Output: A single integer f, if it is optimal for Joe to switch classes f and f+1. If switching no classes is at least as good as the best possible switch, output -1. If there are multiple possible f, output the smallest one.

Sample input:

4

5 3 5 2

Sample output:

2

Explanation: If classes 2 and 3 are switched, then Joe doesn’t need to walk up any stairs during the day. This is clearly optimal.

**The Special Integer Function (12 points)**

For each positive integer n, denote by d(n) the number of positive divisors of n. A positive integer n is said to be “special” if there is no k < n with d(k) = d(n). Given an integer N, compute the sum of all special integers no greater than N.

Input: a single integer N

Output: a single integer, the sum of all special integers no greater than N

Constraints: 1 <= N <= 100000

Sample input: 4

Sample output: 7

Explanation: 3 is not special since d(2) = d(3) = 2. 1, 2 and 4 are special so the answer is 1+2+4=7.

[*Julia’s note: I was expecting one of the test cases to be 100000, in which case it will need to run in a sensible amount of time as well. This pretty much demands you find a way to do it in linear time, i.e. O(N). My solution is, and is sub-second on N=10^5]*

**The Triangle that Crossed the Line (18 points)**

Given a set of N lines in the coordinate plane, y = m\_ix + b\_i , compute the number of triples of these lines that form a triangle that intersects the y axis.

Input: The first line contains a single integer N. N lines follow, the i-th of which contains two space-separated integers m\_i, b\_i, representing the i-th line y=m\_ix + b\_i.

Output: A single integer, the number of triangles formed by the N lines that intersect the y-axis.

Constraints: 1 <= N <= 200000, |m\_i|, |b\_i| <= 109. It is guaranteed that no two lines intersect on the y-axis *[to avoid ambiguity about whether a triangle with such a pair counts or not – Julia]* or have the same slope. In other words, m\_i =/= m\_j and b\_i =/= b\_j for all I =/= j.

Sample input:

4

2 0

-3 10

0 4

5 5

Sample output:

3

Explanation: The triple of lines (1,2,3) do not form a triangle that intersects the y-axis, but the triples (1,2,4), (1,3,4) and (2,3,4) all do.

*Julia’s note: Recall some coordinate geometry: for lines I and j, they meet at x = (b\_i-b\_j)/(m\_i-m\_j). For a triple of lines, you get x\_i, x\_j and x\_k as the x-coordinates of the three corners. You then need two of these to lie on opposite sides of the y-axis (you know you won’t have one that lies exactly on it) – i.e. have opposite sign. Therefore, if any of x\_ix\_j, x\_ix\_k or x\_jx\_k is negative, score the triangle as a hit.*

**Digit Words (5 points in section a and 5 points in section b)**

A digit word is a word where, after possibly removing some letters, you are left with one of the single digits:

ONE, TWO, THREE, FOUR, FIVE, SIX, SEVEN, EIGHT, or NINE

For example:

BOUNCE and ANNOUNCE are digit words since they contain the digit ONE

ENCODE is not a digit word, even though it contains an O, N and E, since they are not in order.

[a] Write a program which reads in a single upper-case word (with at most fifteen letters) and determines if it is a digit word. If the word is not a digit word you should output the word NO. If the word is a digit word, you should output the digit it contains, as a number. You will not be given any words which contain more than one digit.

[b] The made-up word TWFOUR contains the digit TWO and the digit FOUR. What is the length of the shortest made-up word which contains all the digits ONE to FIVE? How about ONE to NINE?

**Sudoku (15 points plus 3 bonus points for the fastest program)**

Sudoku puzzles are designed to be solved by human players with pencil and paper, but with well-defined algorithms it is possible to be solved in almost real time by computer or even smartphone. A standard Sudoku puzzle contains a grid which consists of 9 rows and 9 columns, which can be filled with the digits from 1 to 9. No row, column or major 3x3 box will have any number repeated.

For example, with the starting position:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 5 | 3 |  |  | 7 |  |  |  |  |
| 6 |  |  | 1 | 9 | 5 |  |  |  |
|  | 9 | 8 |  |  |  |  | 6 |  |
| 8 |  |  |  | 6 |  |  |  | 3 |
| 4 |  |  | 8 |  | 3 |  |  | 1 |
| 7 |  |  |  | 2 |  |  |  | 6 |
|  | 6 |  |  |  |  | 2 | 8 |  |
|  |  |  | 4 | 1 | 9 |  |  | 5 |
|  |  |  |  | 8 |  |  | 7 | 9 |

The solution would be

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 5 | 3 | 4 | 6 | 7 | 8 | 9 | 1 | 2 |
| 6 | 7 | 2 | 1 | 9 | 5 | 3 | 4 | 8 |
| 1 | 9 | 8 | 3 | 4 | 2 | 5 | 6 | 7 |
| 8 | 5 | 9 | 7 | 6 | 1 | 4 | 2 | 3 |
| 4 | 2 | 6 | 8 | 5 | 3 | 7 | 9 | 1 |
| 7 | 1 | 3 | 9 | 2 | 4 | 8 | 5 | 6 |
| 9 | 6 | 1 | 5 | 3 | 7 | 2 | 8 | 4 |
| 2 | 8 | 7 | 4 | 1 | 9 | 6 | 3 | 5 |
| 3 | 4 | 5 | 2 | 8 | 6 | 1 | 7 | 9 |

For any inputted starting configuration, write a progam that will produce a solution *[assuming one exists – Julia]*. Note that being efficient is an advantage as the program which processes and solves our test cases in the least amount of time will receive a bonus of 3 points.