Northwestern European Regional Contest 2019

NWERC 2019 Practice Session

Eindhoven, November 16

Practice Problems

A Account Numbers
B Brinkmanship
C Circus

Do not open before the contest has started.

This page is intentionally left blank.

Problem A
Account Numbers
Time limit: 1 second

Organising programming contests is a lot of work. One of the many tasks, some say the most important one, is to arrange reimbursement for the jury (and other volunteers) for their travel to the contest. In order to obtain their reimbursements, the jury members provide (among other things) their *International Bank Account Number*s (IBANs), uniquely identifying their bank accounts. If they make a typo when writing their IBAN, the reimbursement cannot be sent, but fortunately an IBAN contains two check digits that can be used to mitigate such errors. Let us help the contest organisers by verifying that a provided IBAN is correct, so that the reimbursements are not delayed more than necessary.

A Dutch "bank" in Eindhoven by Maarten van Maanen on flickr, cc by-sa

For those of you who do not know, an IBAN is a string consisting of between 15 and 34 upper-case letters and digits. To validate a given IBAN, the following amazing procedure is used.

1. Move the first four characters to the end of the string.
2. Replace each letter by digits, where $A = 10$, $B = 11$, ..., $Z = 35$.
3. Interpret the resulting string as a decimal number and compute the remainder modulo 97. An IBAN is valid if and only if the remainder is 1.

For example, consider the IBAN "`NL20INGB0001234567`". After performing the first step we get the string "`INGB0001234567NL20`", and then after replacing letters with digits we get the number 182316110001234567232120. The remainder of this number modulo 97 equals 1, so this was indeed a valid IBAN.

Input

The input consists of:

- One line with a string s ($15 \leq \text{length}(s) \leq 34$): the IBAN to validate. The IBAN only contains upper case letters and digits.

Output

If the given IBAN is valid, output "`correct`". Otherwise, output "`incorrect`".

Sample Input 1	Sample Output 1
`NL20INGB0001234567`	`correct`

Sample Input 2	Sample Output 2
`NL20ASNB0001234567`	`incorrect`

This page is intentionally left blank.

Problem B
Brinkmanship
Time limit: 5 seconds

You and your colleague are Britain's top negotiators and as such you are in charge of delivering Brexit on time and under budget, as previously agreed.

As usual, the negotiations take place topic-by-topic. Each resolution of a topic takes the world some number of steps either towards Brexit, or away from Brexit, depending on the topic.

Topics are discussed and resolved in one session over the course of a day. There are (non-cyclic) dependencies between some topics. You just need to ensure that for any topic discussed, all of its dependencies are discussed too—then the EU can handle the lowly scheduling work.

You understand that even with the best negotiators in the world, leaving may be impossible. When conferring with other strong political thinkers at your local pub, *Ye Olde Trip to Elea*, your co-patron Zeno argued that no matter how close Brexit appears, it is still a moving target: should the original goals be met tomorrow, the goalposts would already have moved.

Now you are only concerned about bringing Brexit as close as possible after the first day, and leaving the philosophy up to someone else. Choose which topics to resolve in such a way that the total distance negotiations have moved toward the goal by the end of the day is maximised.

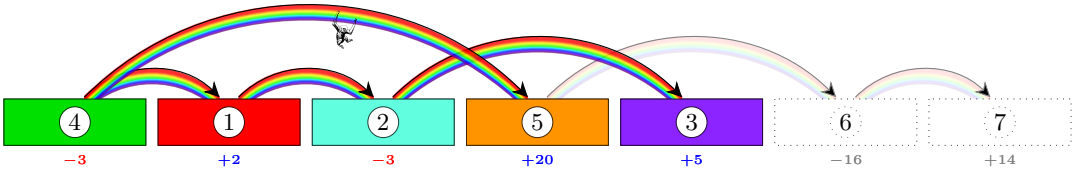

Figure B.1: Illustration of your co-negotiator, and the set of topics they choose to resolve in a solution to Sample Input 2 in order to move 21 steps closer to Brexit.

Input

The input consists of:

- One line containing an integer n ($1 \leq n \leq 500$), the number of topics to be discussed. The topics are numbered from 1 to n.
- n lines, describing the topics. The ith such line starts with two integers x_i and d_i ($-10^6 \leq x_i \leq 10^6$, $0 \leq d_i < n$), the distance that resolving topic i moves us towards Brexit (positive for closer, or negative for further away) and the number of other topics that must be dealt with before topic i can be discussed.

  The remainder of the line has d_i distinct integers $b_{i,1}, \ldots, b_{i,d_i}$ ($1 \leq b_{i,j} \leq n$ and $b_{i,j} \neq i$ for each j), the list of topics that need to be completed before topic i.

It is guaranteed that there are no cycles in the topic dependencies.

Output

Output the maximum possible total distance that can be moved towards Brexit, if an optimal selection of topics is made according to the above rules.

Next, output all of the topics you and your colleague will resolve. If there are multiple optimal solutions, you may output any one of them.

Sample Input 1

```
4
-3 0
5 2 1 3
2 1 4
10 0
```

Sample Output 1

```
14
4 3 2 1
```

Sample Input 2

```
7
2 1 4
-3 1 1
5 1 2
-3 0
20 1 4
-16 1 5
14 1 6
```

Sample Output 2

```
21
5 4 3 2 1
```

Sample Input 3

```
1
-100 0
```

Sample Output 3

```
0
```

Problem C
Circus
Time limit: 5 seconds

In the modern world, the spotlight has shifted entirely from live shows to televised recordings. Well, not entirely... One small troupe of indomitable entertainers still holds out and puts on regular circus performances.

The shows are extremely popular. Streaming media services nearby have caught on and, of course tried to cash in by opening their own circuses. However, they still need good acts to pull in crowds—and the natural solution has been to sneak into the big top to poach ideas and talent.

This can not go on. The ringmaster, inspired by his distant cousins from a small village in Armorica, proposes to build a circular wall around the big top to prevent unauthorised entry.

Figure C.1: Diagram of the indomitable circus, fencing, and the surrounding encampments of Flixium, Fundibulum, Maximillian, and Hulum. This image was adapted from an illustration of the Goseck circle by Kenny Arne Lang Antonsen.

To build this wall, enough fencing will be needed to cover a length equal to the perimeter of the circular tent. How many metres will you need to obtain?

Input

The input consists of:

- One line with an integer a ($1 \leq a \leq 10^{18}$), the area of the circus tent in square metres.

Output

Output the total length of fence needed for the circus palisade wall, in metres. Your answer should have an absolute or relative error of at most 10^{-6}.

Sample Input 1	Sample Output 1
64	28.3592616145

Sample Input 2	Sample Output 2
1234	124.526709336

This page is intentionally left blank.