Northwestern Europe Regional Contest 2020

NWERC 2020 Practice Session

March 27, 2021

Practice Problems

A Another Eruption
B Broken Gearbox
C Cheating

Do not open before the contest has started.

This page is intentionally left blank.

Problem A
Another Eruption

A volcano has recently erupted in Geldingadalur, Iceland. Fortunately this eruption is relatively small, and—unlike the infamous Eyjafjallajökull eruption—is not expected to cause delayed international flights or global outrage.

Geldingadalur eruption by Jon Schow, CC BY-SA

There is some concern about the magmatic gas that has been released as part of the eruption, as it could pose danger to human populations in the surrounding area. Scientists have estimated the total amount of gas emitted, which, due to lack of wind, has spread out uniformly across a circular area around the centre of the volcano. The authorities have evacuated the area, and would now like to close it off by surrounding the perimeter with barrier tape.

Input

The input consists of:

- One line with an integer a ($1 \leq a \leq 10^{18}$), the total area covered by gas in square metres.

Output

Output the total length of barrier tape needed to surround the area covered by gas, in metres. Your answer should have an absolute or relative error of at most 10^{-6}.

Sample Input 1	Sample Output 1
50	25.066282746

Sample Input 2	Sample Output 2
1234	124.526709336

This page is intentionally left blank.

Problem B
Broken Gearbox

The mechanical Turk was an 18th-century sham robot designed to give the illusion of artificial intelligence by playing chess. More importantly, the Turk inspired us to make our own fake robot that appears to solve programming contest problems.

Several interlocking gears by Thomas Claveirole, CC BY-SA

We put some gears on axles inside an uncovered panel to make our machine look more realistic. Since the gears are just for visual effect, they have been placed so as to create an impressive meshing pattern; however, the placement is without any regard to gear ratios or turning direction, so it's possible that none of the gears can really move.

It is guaranteed that every axle was connected to every other axle by meshing, either directly or indirectly. Two axles a and b are said to have directly meshing gears if their distance from one another is equal to the sum of the radii of their gears ($d = r_{g_a} + r_{g_b}$).

Sadly, the gears fell off our machine. We think we collected all of them back up again, but now we're not sure which should go back on which axle. Please find a way of putting the gears on the axles to make them mesh the way they did originally.

Input

The input consists of:

- One line with an integer n ($2 \leq n \leq 10^5$), the number of gears and axles.
- One line with n integers $r_1, \ldots, r_n$ ($2 \leq r_i \leq 10^8$ for each i), the radius of each gear.
- One line with an integer m ($n - 1 \leq m \leq 10^5$), the number of pairs of gear axles that originally had meshing gears.
- m lines, the ith of which contains three integers a_i, b_i and d_i ($1 \leq a_i < b_i \leq n$, $1 \leq d_i \leq 10^8$), the indices of two connected gear axles and the distance between them.

Output

If it is possible to fix the machine with the given gears, output n integers $g_1, \ldots, g_n$, where g_i is the 1-based index of the gear to be put on the ith axle.

Otherwise, output "`impossible`".

Sample Input 1

```
4
20 10 5 15
4
1 2 15
2 3 30
3 4 35
1 4 20
```

Sample Output 1

```
3 2 1 4
```

Sample Input 2

```
6
10 55 80 5 60 50
6
2 4 60
3 4 65
4 5 90
1 6 65
5 6 85
1 4 70
```

Sample Output 2

```
5 6 2 1 3 4
```

Sample Input 3

```
4
1 2 1 1
4
1 4 3
2 4 3
3 4 3
1 3 2
```

Sample Output 3

```
3 1 4 2
```

Sample Input 4

```
3
3 4 5
2
1 2 7
1 3 7
```

Sample Output 4

```
impossible
```

Problem C
Cheating

The Northwestern Europe Regional Contest is coming up soon, and you do not feel confident about your programming skills. In an act of desperation you decide to put on your black hat. After doing some reconnaissance on the contest website you stumble upon a hidden page, guarded by a password prompt in a system known as the *Incredibly Clever Password Control* (ICPC). Perhaps this is where the jury stores the solutions to all the contest problems?

Password by geralt, Pixabay

After inspecting the system you make the following observations about ICPC:

- It has a very strict password policy, requiring the password to consist of exactly four lowercase letters.
- It allows at most 50 incorrect login attempts before users are blocked from the system.
- After every incorrect login attempt, ICPC reports back two numbers about the password that was entered:
  - the number of positions that were correct, and
  - the number of letters that were correct but in an incorrect position.

Use this information to log into the system.

Interaction

This is an interactive problem. Your submission will be run against an *interactor*, which reads the standard output of your submission and writes to the standard input of your submission. This interaction needs to follow a specific protocol:

Your submission must repeatedly send a line consisting of 4 lowercase letters, which represents a login attempt. The interactor then responds with one of:

- The string "`correct`", indicating that the login attempt was successful, and your submission should exit.
- Two integers a and b ($0 \leq a < 4$, $0 \leq b \leq 4$, $a + b \leq 4$), the number of positions in the entered password that were correct, and the number of letters that were correct but in an incorrect position.

Make sure you flush the buffer after each write.

A testing tool is provided to help you develop your solution.

Sample Interaction 1

Read	Write
	`accp`
`1 2`	
	`cacp`
`0 3`	
	`ocpc`

```
3 0
```

```
icpc
```

```
correct
```