课后作业

1. 邮局选址问题

证明：其中一个到数轴上给定的N个点的距离之和最小的点为这N个点的坐标的中位数。

2. 方根

给定正整数A（$1 \le A \le 10^{18}$）和N（$1 \le N \le 60$）。若A是一个N次方数，输出$\sqrt[N]{A}$，否则输出-1。

输入样例：

```
2147483648 31
```

输出样例：

```
2
```

本题的限制为：不得使用 pow()函数。

3. 下一个排列

给定正整数N（$1 \le N \le 100$），以及$1 \dots N$的一个排列$p_1, p_2, \dots, p_N$。求字典序大于$p_1, p_2, \dots, p_N$的$1 \dots N$的最小排列。若不存在这样的排列，输出-1。

输入样例 1：

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

输出样例 1：

```
2 4 1 5 3
```

输入样例 2：

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

输出样例 2：

```
2 5 1 3 4
```

4. (USACO 2013 December - Bronze 3) Wormholes

[Brian Dean, 2013]

Farmer John's hobby of conducting high-energy physics experiments on weekends has backfired, causing N wormholes (2 <= N <= 12, N even) to materialize on his farm, each located at a distinct point on the 2D map of his farm.

According to his calculations, Farmer John knows that his wormholes will form N/2 connected pairs. For example, if wormholes A and B are connected as a pair, then any object entering wormhole A will exit wormhole B moving in the same direction, and any object entering wormhole B will similarly exit from wormhole A moving in the same direction. This can have rather unpleasant consequences. For example, suppose there are two paired wormholes A at (0,0) and B at (1,0), and that Bessie the cow starts from position (1/2,0) moving in the +x direction. Bessie will enter wormhole B, exit from A, then enter B again, and so on, getting trapped in an infinite cycle!

Farmer John knows the exact location of each wormhole on his farm. He knows that Bessie the cow always walks in the +x direction, although he does not remember where Bessie is currently located. Please help Farmer John count the number of distinct pairings of the wormholes such that Bessie could possibly get trapped in an infinite cycle if she starts from an unlucky position.

PROBLEM NAME: wormhole

INPUT FORMAT:

* Line 1: The number of wormholes, N.

* Lines 2..1+N: Each line contains two space-separated integers describing the (x,y) coordinates of a single wormhole. Each coordinate is in the range 0..1,000,000,000.

SAMPLE INPUT (file wormhole.in):

```
4
0 0
1 0
1 1
0 1
```

INPUT DETAILS:

There are 4 wormholes, forming the corners of a square.

OUTPUT FORMAT:

* Line 1: The number of distinct pairings of wormholes such that Bessie could conceivably get stuck in a cycle walking from some starting point in the +x direction.

SAMPLE OUTPUT (file wormhole.out):

```
2
```

OUTPUT DETAILS:

If we number the wormholes 1..4, then by pairing 1 with 2 and 3 with 4, Bessie can get stuck if she starts anywhere between (0,0) and (1,0) or between (0,1) and (1,1). Similarly, with the same starting points, Bessie can get stuck in a cycle if the pairings are 1-3 and 2-4. Only the pairings 1-4 and 2-3 allow Bessie to walk in the +x direction from any point in the 2D plane with no danger of cycling.

5. (USACO 2012 November - Bronze 3) Horseshoes

[Brian Dean, 2012]

Although Bessie the cow finds every string of balanced parentheses to be aesthetically pleasing, she particularly enjoys strings that she calls "perfectly" balanced -- consisting of a string of ('s followed by a string of)'s having the same length. For example:

```
(((())))
```

While walking through the barn one day, Bessie discovers an N x N grid of horseshoes on the ground, where each horseshoe is oriented so that it looks like either (or). Starting from the upper-left corner of this grid, Bessie wants to walk around picking up horseshoes so that the string she picks up is perfectly balanced. Please help her compute the length of the longest perfectly-balanced string she can obtain.

In each step, Bessie can move up, down, left, or right. She can only move onto a grid location containing a horseshoe, and when she does this, she picks up the horseshoe so that she can no longer move back to the same location (since it now lacks a horseshoe). She starts by picking up the horseshoe in the upper-left corner of the grid. Bessie only picks up a series of horseshoes that forms a perfectly balanced string, and she may therefore not be able to pick up all the horseshoes in the grid.

PROBLEM NAME: hshoe

INPUT FORMAT:

* Line 1: An integer N (2 <= N <= 5).

* Lines 2..N+1: Each line contains a string of parentheses of length N. Collectively, these N lines describe an N x N grid of parentheses.

SAMPLE INPUT (file hshoe.in):

```
4
(())
()((
(()(
))))
```

OUTPUT FORMAT:

* Line 1: The length of the longest perfectly balanced string of horseshoes Bessie can collect. If Bessie cannot collect any balanced string of horseshoes (e.g., if the upper-left square is a right parenthesis), output 0.

SAMPLE OUTPUT (file hshoe.out):

```
8
```

OUTPUT DETAILS:

The sequence of steps Bessie takes to obtain a balanced string of length 8 is as follows:

```
1())
2)((
345(
876)
```

6. (USACO 2019 January - Silver 2) Icy Perimeter

Farmer John is going into the ice cream business! He has built a machine that produces blobs of ice cream but unfortunately in somewhat irregular shapes, and he is hoping to optimize the machine to make the shapes produced as output more reasonable.

The configuration of ice cream output by the machine can be described using an $N \times N$ grid $(1 \leq N \leq 1000)$ as follows:

```
##....
....#.
.#..#.
.#####
...###
....##
```

Each '.' character represents empty space and each '#' character represents a 1×1 square cell of ice cream.

Unfortunately, the machine isn't working very well at the moment and might produce multiple disconnected blobs of ice cream (the figure above has two). A blob of ice cream is connected if you can reach any ice cream cell from every other ice cream cell in the blob by repeatedly stepping to adjacent ice cream cells in the north, south, east, and west directions.

Farmer John would like to find the area and perimeter of the blob of ice cream having the largest area. The area of a blob is just the number of '#' characters that are part of the blob. If multiple blobs tie for the largest area, he wants to know the smallest perimeter among them. In the figure above, the smaller blob has area 2 and perimeter 6, and the larger blob has area 13 and perimeter 22.

Note that a blob could have a "hole" in the middle of it (empty space surrounded by ice cream). If so, the boundary with the hole also counts towards the perimeter of the blob. Blobs can also appear nested within other blobs, in which case they are treated as separate blobs. For example, this case has a blob of area 1 nested within a blob of area 16:

```
#####
#...#
#.#.#
#...#
#####
```

Knowing both the area and perimeter of a blob of ice cream is important, since Farmer John ultimately wants to minimize the ratio of perimeter to area, a quantity he calls the icyperimetric measure of his ice cream. When this ratio is small, the ice cream melts slower, since it has less surface area relative to its mass.

INPUT FORMAT (file perimeter.in):

The first line of input contains N, and the next N lines describe the output of the machine. At least one '#' character will be present.

OUTPUT FORMAT (file perimeter.out):

Please output one line containing two space-separated integers, the first being the area of the largest blob, and the second being its perimeter. If multiple blobs are tied for largest area, print the information for whichever of these has the smallest perimeter.

SAMPLE INPUT:

```
6
##....
....#.
.#..#.
.#####
...###
....##
```

SAMPLE OUTPUT:

```
13 22
```

Problem credits: Brian Dean