

Freebit Question – Hello World

(30 points)

Introduction

The contest has started!

Let's test everything first.

Is the internet setting correct?

Is the answer submit working well?

Did you use STDOUT to output your answers?

If everything is ready, 30 points are yours! Let's... Go! Go! Go!

Description

Please write a program to output Hello World!

Input Format

This problem requires no input.

Output Format

[A~Z][a~z], space, and common english punctuation.

Data Range

English [A~Z][a~z], space, and exclamation mark "!".

Data Examples

Input Example 1

(no input)

Output Example 1

Hello World!

Example Explanation

Input Example 1 has no input, simply output Hello World!

Question 1 - PrimeNumber

(5 points)

Description

The definition of a prime number is a number that is divisible only by itself and 1 for all of Natural Numbers greater than 1.

Please write a program that determines whether the input number N is a prime number.

If N is a prime number, please output all prime numbers less than or equal to N, including N.
if N is not a prime number, please output all non-prime numbers less than or equal to N, including N. (including 1)

Input Format

N

N is a positive integer.

Output Format

If N is a prime number, please output all prime numbers less than or equal to N, including N.
if N is not a prime number, please output all non-prime numbers less than or equal to N, including N. (including 1)
Please sort the result set in descending.

Data Range

$1 < N \leq 5000$

Data Examples

Input Example 1

10

Output Example 1

10 9 8 6 4 1

Input Example 2

15

Output Example 2

15 14 12 10 9 8 6 4 1

Input Example 3

23

Output Example 3

23 19 17 13 11 7 5 3 2

Example Explanation

The input of Example 1 or 2 is not a prime number, so the output will be all non-prime numbers less than or equal to itself.

Example 3, the input 23 is a prime number, so the output will be all those prime numbers less than or equal to itself.

Question 2 - MaxSquare

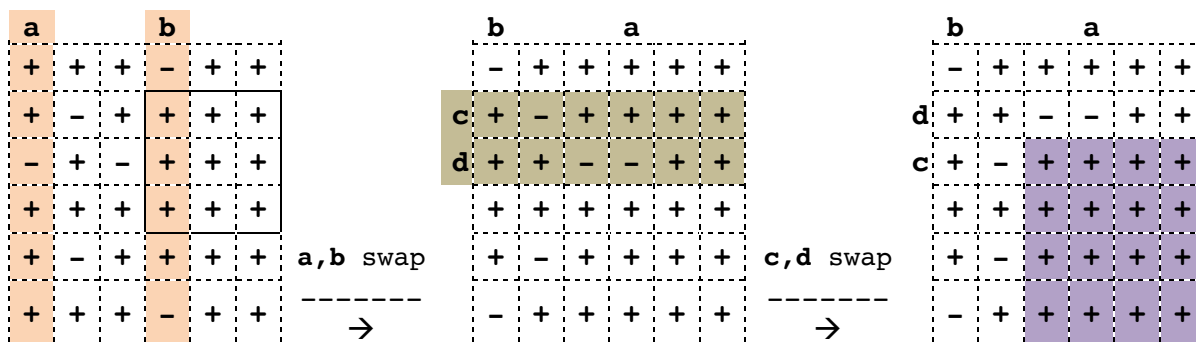
(20 points)

Description

For a square table (same number of rows and columns), every element status is either + or -. A square sub-table that have all + elements is called a positive square sub-table. As shown in the picture below, within the 6x6 square table, the marked area is a 3x3 positive square sub-table.

+	+	+	-	+	+
+	-	+	+	+	+
-	+	-	+	+	+
+	+	+	+	+	+
+	-	+	+	+	+
+	+	+	-	+	+

The "Maximum Positive Value" is defined as: the side length of the biggest positive square sub-table after swap any times of rows and columns. For each swap, a whole complete row (or column) will swap with another whole complete row (or column).



In the example above, the original 3x3 positive square sub-table will go to the 4x4 positive square sub-table (the biggest possible) after swap column "a" with column "b", then swap row "c" with row "d". Therefore, the Maximum Positive Value for this square table is 4 (after 2 swaps).

Please write a program to calculate the Maximum Positive Value of a square table. Also, please tell us the minimum number of swaps required to obtain the biggest positive square sub-table.

Input Format

1. The first line is a positive number N. It is the side length of the square table. Both width and height have N elements.

2. Followed by N rows, each with N characters (+ or -), representing the status of elements.

Output Format

2 integers separated by a comma. The first integer represents the Maximum Positive Value, and the second integer represents the minimum number of swaps required to obtain the biggest positive square sub-table.

Data Range

N is a positive integer, not over 20.

Data Example

Input Example 1

```
6
+++--+
+-++++
-+-+++
++++++
+-++++
+++--+
```

Output Example 1

```
4, 2
```

Example Explanation 1

As shown in the description, after 2 swaps (row and column 1 each), a 4x4 maximum positive square sub-table can be obtained. Therefore, the output is 4,2 (Maximum Positive Value, minimum swap number).

Input Example 2

```
2
--
-+
```

Output Example 2

```
1, 0
```

Example Explanation 2

The max positive square sub-table is 1x1 and no swap is needed.

Input Example 3

```
19
++-+-++-++++-+++++++
+-++++-+++++++-+-+
-+++++++-+-+++++++
+++++++-+-++-
```

```

++++-++++++-+++++
+++++-++++++
++++++-+++++
++++-+++++-+
+-+--++++++
+++-++++-+-++++
--++++-+-++-+
-+-++++-+-++-+
++++-+++++-+-+
++-+-++++-++++
+-+-++++-+-++-+
++++++-++++-
+++++-+-+++++
++++++-++++-+
+++++-+++++

```

Output Example 3

9,6

Example Explanation 3

After 6 swaps, the biggest 9x9 positive square sub-table will be obtained.