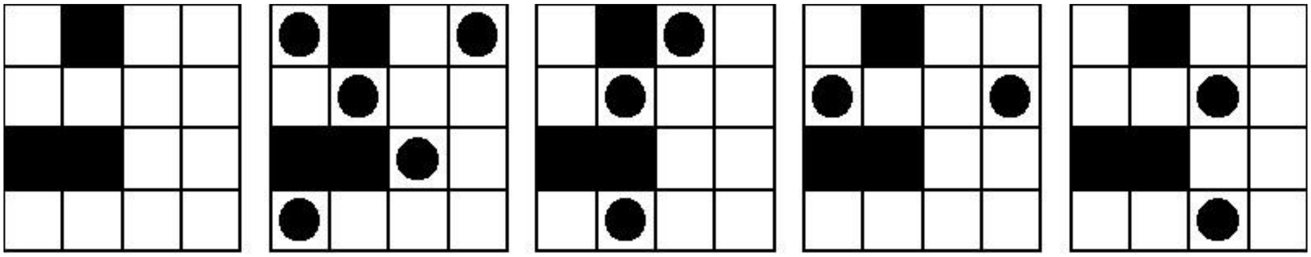
"KamaChallenge-2015"

A. Rook

Chesspiece named rook can move vertical and horizontal. In this problem there would be square parts of a chessboard at most 4x4, all the cells of it are white and there are walls making rook impossible to pass. Your task is to find the largest number of rooks to be put on the part of a chessboard in order to have all the rooks unreachable to each other in one move. The following five pictures show different situations about the rooks. The first picture has no rooks. The second and the third pictures has the rooks unreachable to each other in one move, the 4th and the 5th show situations that don’t fulfill the rule of one move. For the part of a chessboard on the following picture there are different solutions that gives the answer 5.



You should write a program that prints the largest number of rooks to be put on the part of a chessboard in order to have all the rooks unreachable to each other in one move for the situation from the input.

Input file

The first line of input file contains the only number n (n<=4) – the length of the side of the part. The next n lines describe the situation. Symbol "." describes empty space, "X" is for the wall. **Output file**

The only line of output file should contain the largest number of rooks to be put on the part of a chessboard in order to have all the rooks unreachable to each other in one move. **Sample input**

```
4
.X..
....
XX.. ....
```

Sample output 5

B. Self numbers

In 1949 Indian mathematiciam D.R. Kaprekar discovered a class of numbers called self numbers. For every positive integer n we define a function d(n) as n added to the sum of digits of n. For example, d(75)=75+7+5=87. If some positive integer n is taken, we can define the infinite sequence n, d(n), d(d(n)), d(d(d(n))),... If we start from 33, the next number would be 33+3+3=39, after it 39+3+9=51, then 51+5+1=57 so we have the following sequence:

33, 39, 51, 57, 69, 84, 96, 111, 114, 120, 123, 129, 141,...

The number n is called a generator for d(n). In the example above 33 is generator for 39, 39 is generator for 51, 51 is generator for 57 etc. Some numbers has more that one generator, for example for 101 there are two generators: 91 and 100. A number that don’t have a generator called self number. There are 13 self numbers less than 100: 1, 3, 5, 7, 9, 20, 31, 42, 53, 64, 75, 86 and 97.

You should write a program that counts the sum of all the self numbers less than 10000. **Input format**

The only line of input contains the number 10000. **Output format**

In the only line of output your program should print the sum of all the self numbers less than 10000.

C. Stones

There is a set of N stones (1<=N<=10), and every stone has two numbers for it: mass m_i and the price v_i (i=1, 2, ..., N). You should put all the stones in storages A and B with a stonethrower.

A stonethrower works as follows. The stones are put in the same storage until total mass of stones in this storage is larger that in the other storage by D. The a stonethrower switches to another storage. In the beginning all the storages are empty and a stonethrower is set to storage A.

You should write a program that prints the largest possible total mass of stones in storage B at the end. **Input format**

In the first line there are two integer numbers N and D. In the following N lines there are numbers p_i and v_i. All the numbers except N are from 0 to 10000.

Output format

In the only line of output should be the largest possible total mass of stones in storage B at the end. **Sample input**

4 2
2 2
2 2
1 1
1 1

Sample output

3

D. Strings

You are given a string with characters a and b only and length no more than 255 symbols. You can make a string shorter as follows: string a*a or b*b, where * is some non-empty substring can be reduced to *.

You should write a program that find the minimal length of the string reduced from the given one.

Input format

The only line of input file contains the string.

Output format

The only line of output should have the minimal length. **Sample input**

aab

Sample output

3

E. Bridge

A group of N tourists (1<=N<=50) reached old bridge and is going to pass it as soon as possible. The bridge is suitable for at most two people on it. Tourists should go with a flashlight and they have only one, so somebody should return with it. For every tourist there is a time t_i to pass a bridge (i=1..N, 1<=t_i<=100). If two people go by the bridge they spend the most time of two values.

You should write a program to find the least time for a group to go to another side.

Input format

In the first line there is a number N. In the second line there are integer numbers t_i divided by a space.

Output format

The only line of the output should contain a single integer: the minimal time to be on the other side. **Sample input**

4
6 7 6 5

Sample output

29

F. The last digit

You are to write a program that outputs the last non-zero digit of the number of combinations of N elements taken M at a time.

Input format

The only line of input file contains two positive integers N and M (1<=N,M<=1000000), where N>=M.

Output format

The only line of output file should contain the only digit, namely the last non-zero digit of the number of combinations specified by input.

Sample input

```
4 2
```

Sample output

```
6
```