Little Danica was completely disgusted by the Committee's decision to cancel one round of qualifiers this year, and she started breaking everything within reach, including numbers that happened to be nearby. Whenever she sees a pair of natural numbers, she throws one of them at the other so that they collide. When two numbers collide, a new natural number is formed.

When natural numbers X and Y collide, each digit of one is compared with the corresponding digit of the other: the ones digit is compared to the ones digit, the tens digit with the tens, the hundreds with the hundreds, and so on. The lower of the two digits disappears, and the higher is used to form the resulting number. If the digits are equal, both will be used, and the result will have more digits than X and Y. If one of the numbers has fewer digits than the other, the "extra" digits from the other one are always used (even if they are zero).

Danica can't deal with all the numbers she is surrounded by, and asked you for help in colliding them. To do so, you first need to answer a question: which number will be formed when two numbers X and Y collide?

Input format

The first line of standard input contains the first natural number X, and the second line contains the second number Y that participates in the collision, as described in the problem statement.

Output format

Your program should print one line to the standard output: the newly-formed number that is the result of colliding X and Y.

Sample 1

Input

```
73
28
```

Output

```
78
```

Sample 2

Input

```
64
357
```

Output

```
367
```

Sample 3

Input

```
234
135
```

Output

```
2335
```

Sample 4

Input

```
99099
9999
```

Output

```
99999999
```

Sample 5

Input

```
2
100
```

Output

```
102
```

Explanation

In the first testcase, numbers 73 and 28 collide. Both have two digits, and the result will have the greater of the tens digits 7 and 2 as its tens digit, and the greater of 3 and 8 as the ones digit. The result is therefore 78, since 7 is greater than 2 and 8 is greater than 3.

In the second testcase, numbers 64 and 357 collide. One has three digits, and the other has two, so the hundreds digit will be the 3 taken from 357. The remaining digits are handled as in the previous case: 6 is greater than 5, and 4 is greater than 7, so the result will be 367.

In the third testcase, numbers 234 and 135 collide. The matching digit pairs are 2 and 1, 3 and 3, and 4 and 5. The hundreds and ones digits are the larger ones in the corresponding pairs: 2 and 5. In the tenths place the two digits are equal and will both be included in the result, which is then 2335.

In the fourth testcase, numbers 99099 and 9999 collide. As one has five digits and the other four, the lower four digit pairs will be compared, and the topmost digit of 99099 will be included in the result by default. There are only two relevant cases: a pair of 9 and 0, that produces a single 9 in the result, and a pair of 9 and 9, which results in 99. The result is then 99999999, since the two numbers contain a total of eight nines, which will all be included in the result.

Finally, in the fifth testcase, numbers 2 and 100 collide. The digits 1 and 0 are simply copied from the longer number, and the ones digit is 2, since it is greater than the corresponding 0, making the result 102.

Constraints

- $1 \leq X, Y \leq 10^9$

Testcases are split into five disjoint groups:

- In tests worth 10 points: $1 \leq X, Y < 10$, that is, X and Y have one digit each.
- In tests worth 15 points: $10 \leq X, Y < 100$, that is, X and Y have two digits each.
- In tests worth 15 points: $100 \leq X, Y < 1000$, that is, X and Y have three digits each.
- In tests worth 20 points: $10^3 \leq X, Y \leq 10^9$ and X and Y have the same number of digits.
- In tests worth 40 points: no additional constraints.

Note

The numbers given in the input will not have leading zeroes. Your output should not contain leading zeroes.