Lexicographical Order
Problem ID: a07p01lexicographicalorder

Write a function, `precedes(one_string, another_string)`, that receives two strings as parameters. The function returns the string that comes first in alphabetical order. For example, Hungry comes before Starving alphabetically, because h comes before s, and Duchess precedes Duke, because both strings start with du, but c precedes k. The function should ignore case.

Note that we are testing your code differently in this task, please only submit your function definitions, without any code outside the functions! The main python file, which handles input and output, is already provided. You can download and place the main file in the same directory as your python file. You can then run the main python file we provide to try out the samples.

Input

Your function will be called with two given parameters as input. Each parameter will be a string, consisting of 1 to 20 letters from the English alphabet and spaces. You do not need to check this.

As usual, the input is given in an input file, and a description of that follows below. But the given main file will take care of reading the input and passing the parameters to your function, as well as printing the result to the output for you, so you don't really need to worry about that, except for when you want to test your code locally, then you might want to do something similar. The samples given below show the input and corresponding output as they appear in these files.

The input consists of two lines. Each line contains one string, and those are the ones that will be passed as the parameters to the function.

Formally (since we know that mathematical notation is your favourite language), the input consists of $n = 2$ lines, and line i contains a string s_i, for $1 \leq i \leq n$. Further, $1 \leq |s_i| \leq 20$, where $|s|$ denotes the length of a string s.

Output

The output of the function should be a string, the same string as one of the input strings, the one that precedes the other alphabetically. The comparison should be made ignoring case, but the result should be the original string unchanged.

This will result in an output file that is one line containing the value returned from calling `precedes`, with the first input string s_1 passed as the first argument to the function, and the second input string s_2 passed as the second argument.

The function `precedes` will be called exactly once, in each execution of the program, which is to say for each test case.

Sample Input 1

```
monkey
cat
```

Sample Output 1

```
cat
```

Sample Input 2

```
muna
gleyma
```

Sample Output 2

```
gleyma
```

Sample Input 3

```
cat
bat
```

Sample Output 3

```
bat
```

Sample Input 4

```
Programming
forritun
```

Sample Output 4

```
forritun
```

Sample Input 5

```
The past
the present
```

Sample Output 5

```
The past
```

Sample Input 6

```
Duke
Duchess
```

Sample Output 6

```
Duchess
```

Sample Input 7

```
President
President
```

Sample Output 7

```
President
```

Sample Input 8

```
begin
beginning
```

Sample Output 8

```
begin
```

Sample Input 9

```
beginning
begin
```

Sample Output 9

```
begin
```

Sample Input 10

```
Hungry
Starving
```

Sample Output 10

```
Hungry
```