# Problem 4 – Longest Alphabetical Word

Nakov enjoys playing with words. Recently he invented the following puzzle game. He starts by given word **w** (e.g. "**softwareuniversity**") and he fills a square block of size **n**\***n** (e.g. n=7) with this word as many times as it fits, from left to right and from up to down (see the example on the right). It is also called Nakov's square block of word **w** and size **n**.

softwar

euniver

sitysof

tw**a**reun

iversit

ysoftwa

reunive

Nakov defines an **alphabetical word** as a sequence of letters, where each letter is alphabetically after its previous letter in the word. For example, "abc", "fo" and "aeou" are alphabetical words, but "zabc", "srevi" and "ntaeou" are not.

Now Nakov wants to find the **longest alphabetical word** in the obtained square block. The word can start anywhere in the square block and can run in left, right, up or down direction and cannot go outside of the square block. In our example, if we start from row 3 and column 2 in our 7 x 7 square block, we find the following alphabetical words: "**aw**" (left direction), "**ar**" (right direction), "**at**" (up direction) and "**aeou**" (down direction).

Write a program that reads a word **w** and a number **n** and finds the **longest alphabetical word** in Nakov's square block of word **w** and size **n**. If more than one longest alphabetical words exist in the block, find the smallest of them in the standard lexicographical order.

## Input

The input data should be read from the console. The input data consists of **exactly two lines**:

* The first line will hold the word **w**.
* The second line will hold the size **n**.

The input data will always be valid and in the format described. There is no need to check it explicitly.

## Output

You have to print at the console the **longest alphabetical word**.

## Constraints

* The word **w** will be a non-empty string, consisting of lower Latin letters, up 1000.
* The size of the square **n** will be an integer value in the range [1…50].
* Allowed work time for your program: 0.25 seconds.
* Allowed memory: 16 MB.

## Examples

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | **Output** | **Block** |  | **Input** | **Output** | **Block** |  | **Input** | **Output** | **Block** |
| softwareuniversity  7 | aeou | softwar  euniver  sitysof  tw**a**reun  iv**e**rsit  ys**o**ftwa  re**u**nive | alpha  6 | ahp | alphaa  l**pha**al  phaalp  haalph  aalpha  alphaa | java  3 | aj | **j**av  **a**ja  vaj |