



itertools.combinations_with_replacement() *

Your itertools.combinations_with_replacement() submission got 10.00 points.

X

Try the next challenge | Try a Random Challenge

Problem

Submissions

Leaderboard

Editorial A

itertools.combinations_with_replacement(iterable, r)

This tool returns \boldsymbol{r} length subsequences of elements from the input iterable allowing individual elements to be repeated more than once.

Combinations are emitted in lexicographic sorted order. So, if the input iterable is sorted, the combination tuples will be produced in sorted order.

Sample Code

```
>>> from itertools import combinations_with_replacement
>>> print list(combinations_with_replacement('12345',2))
[('1',\ '1'),\ ('1',\ '2'),\ ('1',\ '3'),\ ('1',\ '4'),\ ('1',\ '5'),\ ('2',\ '2'),\ ('2',\ '3'),\ ('2',\ '4'),\ ('2',\ '5'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\ ('3',\ '3'),\
  '4'), ('3', '5'), ('4', '4'), ('4', '5'), ('5', '5')]
>>> A = [1,1,3,3,3]
>>> print list(combinations(A,2))
[(1, 1), (1, 3), (1, 3), (1, 3), (1, 3), (1, 3), (1, 3), (3, 3), (3, 3), (3, 3)]
```

Task

You are given a string ${\it S}$.

Your task is to print all possible size ${\pmb k}$ replacement combinations of the string in lexicographic sorted order.

Input Format

A single line containing the string \boldsymbol{S} and integer value \boldsymbol{k} separated by a space.

Constraints

$0 < k \leq len(S)$

The string contains only UPPERCASE characters.

Output Format

Print the combinations with their replacements of string \boldsymbol{S} on separate lines.

Sample Input

HACK 2

Sample Output

AC

ΑН

AK CC

СН

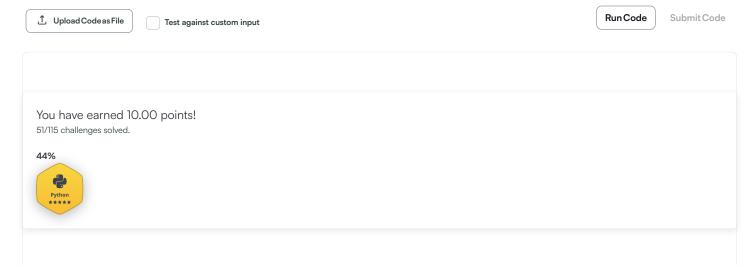
CK

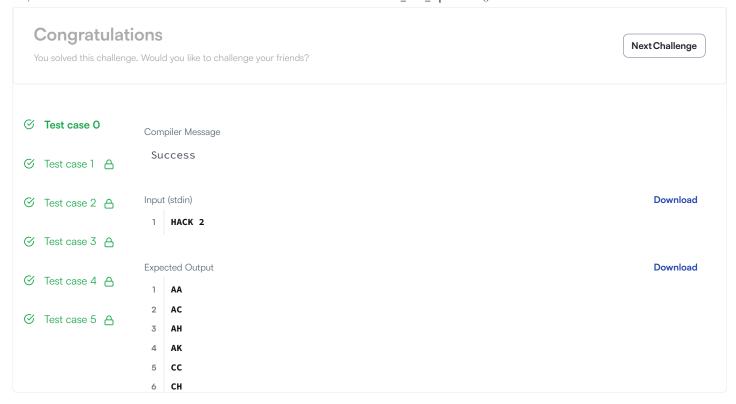
НН

HK KK

```
Change Theme Language Python 3
                                                                                                 ૱ :
    # Enter your code here. Read input from STDIN. Print output to STDOUT
1
2
    from itertools import combinations_with_replacement
4
    string, k = input().split()
5
    string = sorted(string)
    k = int(k)
    lst = list(combinations_with_replacement(string, k))
9
10
    for j in lst:
        print("".join(j))
11
12
13
```

EMACS Line: 13 Col: 1





Blog | Scoring | Environment | FAQ | About Us | Helpdesk | Careers | Terms Of Service | Privacy Policy