itertools.combinations_with_replacement() ★

20 more points to get your next star!

Rank: **1244976** | Points: **50/70**

Your itertools.combinations_with_replacement() submission got 10.00 points. Share Tweet

You are now 20 points away from the 2nd star for your python badge.

Try the next challenge | **Try a Random Challenge**

Problem Submissions Leaderboard Editorial

itertools.combinations_with_replacement(iterable, r)

This tool returns 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',
>>>
>>> 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 S.

Your task is to print all possible size k replacement combinations of the string in lexicographic sorted order.

Input Format

A single line containing the string S and integer value 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 S on separate lines.

Sample Input

```
HACK 2
```

Sample Output

```
AA
AC
AH
AK
CC
CH
CK
HH
HK
KK
```

Change Theme Language Python 3

```
# Enter your code here. Read input from STDIN. Print output to STDOUT
S = list(input())
k = list()
k.append(S[-1])
S.remove(S[-1])
S = list(S)
S.remove(S[-1])
k = list(map(int, k))
K = [str(integer) for integer in k]
K_string = "".join(K)
res = int(K_string)
S.sort()
from itertools import combinations_with_replacement
#Error function for constraints
while 0 > res > len(S) and any((char.islower()) for char in S):
    print ("Error, please input again:")
    S = list(input())
    k = int(input())
    if 0 < res <= len(S) and all((char.isupper()) for char in S):
        break
#main function
if 0 < res <= len(S) and all((char.isupper()) for char in S):
    for n in list(combinations_with_replacement(S, res)):
        print ("".join(n))
```

Line: 13 Col: 52

Upload Code as File Test against custom input Run Code Submit Code

You have earned 10.00 points!

You are now 20 points away from the 2nd star for your python badge.

43% 50/70

Congratulations

You solved this challenge. Would you like to challenge your friends?

Next Challenge

- **Test case 0**
- Test case 1
- Test case 2
- Test case 3
- Test case 4
- Test case 5

Compiler Message

```
Success
```

Input (stdin) Download

```
1 HACK 2
```

Expected Output Download

```
1 AA
2 AC
3 AH
4 AK
5 CC
```