Permutation of a given string can be done in different ways. In order to get different permutations of a string, when all characters in the string are different, Donald E. Knuth gave the following process.

To get all the permutations of a n character string $(a_1a_2 \ldots a_n)$, using each permutation $a_1a_2 \ldots a_{n-1}$, we can form n others by inserting a_n (n-th character) in all possible places. Thus we get $n!$ permutations of that string.

For example, to generate all permutations of '`ACB`', we first start with '`A`', then insert '`C`' and then insert '`B`'.

Col 1	Col 2	Col3	Permutation Index
A	CA	BCA	1
		CBA	2
		CAB	3
	AC	CBA	4
		ABC	5
		ACB	6

So we see that using above technique, the permutations of '`ACB`' are generated in a particular order. Here 2nd permutation of '`ACB`' is the string '`CBA`' or permutation index of '`CBA`' is 2. In this problem you will be given a string and a permutation index, I. You have to find the I'th permutation of the given string.

Input

First line of the input file will contain an integer denoting the number of test cases to follow. For each test case there will be two lines. First line of each test case will contain a string of length less than or equal to 26. The characters of the string will be all upper case letters and different. Next line will contain a permutation index, I. Range of I is from 1 to $\min(n!, 2^{31} - 1)$, where n is the length of the string.

Output

For each test case, print the I'th permuted string of the given string in a line. Look at Sample input and output for details.

Sample Input

```
4
ACB
2
ABC
1
ABC
6
ABCD
12
```

Sample Output

```
CBA
CBA
ABC
BACD
```