

# Aho-Corasick Trie

The **Aho–Corasick string matching algorithm** is a [string searching algorithm](#) invented by [Alfred V. Aho](#) and [Margaret J. Corasick](#). It is a kind of dictionary-matching algorithm that locates elements of a finite set of strings (the "dictionary") within an input text. It matches all patterns simultaneously.

The Algorithm described above, requires an input file generator. The generator generates a text of length  $L$ , by choosing  $L$  characters randomly. Probability of choosing each character is given as priori, and independent of choosing others. Now, given a set of patterns, calculate the probability of a valid program generating "no".

## Input

First line contains an integer  $T$ , the number of test cases. Each case starts with an integer  $K$ , the number of pattern strings. Next  $K$  lines each contain a pattern string, followed by an integer  $N$ , number of valid characters. Next  $N$  lines each contain a character and the probability of selecting that character,  $p_i$ . Next an integer  $L$ , the length of the string generated. The generated text can consist of only the valid characters, given above. There will be a blank line after each test case.

## Output

For each test case, output the number of test case, and the probability of getting a "no".

## Constraints

- $T \leq 100$
- $K \leq 40$
- Length of each pattern string is between 1 and 20
- Each pattern string consists of only alphanumeric characters ('a' to 'z', 'A' to 'Z', '0' to '9')
- Valid characters are all alphanumeric characters
- $\sum p_i = 1$
- $L \leq 100$

## Example

### Input

```
2
1
a
2
a 0.5
b 0.5
2
```

```
2
ab
ab
2
a 0.2
b 0.8
2
```

**Output:**

Case #1: 0.250000

Case #2: 0.840000