

2. 연산의 다양성

$\{a, b, c\}$ 세 원소로 이루어지는 연산 테이블이 아래 그림과 같이 주어진다. 예를 들어, $ab=b$, $ba=c$, $cc=c$ 와 같이 계산된다. $a(b((cb)a)) = a$ 이다. 괄호를 치는 방법에 따라 연산의 순서가 달라지고 결과도 달라진다. $\{a, b, c\}$ 로 이루어진 임의의 문자열 $x_1x_2 \cdots x_n$ ($x_i \in \{a, b, c\}$)에 대하여 결과가 a, b, c 가 되는 괄호치기 방법의 총 수는 각각 몇 가지인지 계산하는 알고리즘을 작성하라. 예를 들어, 문자열 $abaa$ 의 경우를 보자. 결과가 a 가 되는 괄호치기 방법은 $((ab)a)a$ 한 가지이다. b 가 되는 방법은 $((ab)(aa))$, $(a(ba))a$, $a((ba)a)$, $a(b(aa))$ 4가지이다. c 가 되는 방법은 없다.

	a	b	c
a	b	b	a
b	c	b	a
c	a	c	c

알고리즘은 최대한 효율적으로 작성하라. 10개의 케이스 모두를 수행한 알고리즘의 총 수행 시간이 1초를 넘지 않아야 한다. 채점을 위한 컴파일 시에 최적화 옵션은 쓰지 않는다.

[입력]

입력 파일에는 10 개의 테스트 케이스가 주어진다. 각 케이스는 2 줄로 이루어진다. 첫 줄에는 N 이 주어지고, 다음 줄에는 a, b, c 로 이루어진 문자열(길이는 1부터 30사이)이 주어진다. 입력파일의 이름은 “input2.txt”이다.

[출력]

각 테스트 케이스에 대해서, 케이스의 번호를 “#x” 의 형식으로 출력한 후(여기서 x 는 테스트 케이스 번호), 공백을 하나 둔 다음 주어진 케이스에서 각각 a, b, c 가 되는 경우의 수들을 나열한다. 각 수 사이는 역시 공백으로 구분한다. 출력 결과물을 “output2.txt”로 저장한다.

[예제]

입력 (input2.txt)

2	← 1번 케이스
ac	
3	← 2번 케이스
bbc	
...	

출력 (output2.txt)

#1 1 0 0
#2 1 0 1
...