

순열(Permutation)

[문제] n 개 소문자로 단어가 있다. 예를 들면 banana는 {a, a, a, b, n, n} 6개 소문자로 구성된 문자열 중 하나이다. 이 소문자 중복 집합으로 만들어진 길이 n 인 문자열을 사전식으로 모두 나열한다고 가정해보자. 여러분은 주어진 단어에서 사전식 순서로 볼 때 앞 뒤 $+k$ 번째 $-k$ 번째 문자열을 구해서 출력해야 한다. 예를 들어 {a, a, a, b, n, n}로 구성된 문자열을 사전식 순서를 27번부터 53번까지 나열하면 이래 표와 같다. $k=5$ 이라면 banana에서 앞 뒤로 5번째 순서의 문자열인 **annbaa**과 **bnnaaa**를 출력해야 한다.

만일 k 값이 너무 커 $-k$, $+k$ 번째 해당하는 단어가 없을 경우에는 "none"이라는 문자열을 출력해야 한다. "banana"의 경우 생성 문자열은 총 60개인데 $k=40$ 이면 해당되는 단어는 없다.¹⁾ 만일 $k=2$ 라면 33번째인 baanna와 42번째인 naaanb를 출력해야 한다.

27 :	anbnaa	36 :	bannaa	45 :	naanab
28 :	annaab	37 :	bnaaan	46 :	naanba
29 :	annaba	38 :	bnaana	47 :	nabaan
30 :	annbaa	39 :	bnanaa	48 :	nabana
31 :	baaann	40 :	bnnaaa	49 :	nabnaa
32 :	baanan	41 :	naaabn	50 :	nanaab
33 :	baanna	42 :	naaanb	51 :	nanaba
34 :	banaan	43 :	naaban	52 :	nanbaa
35 :	banana	44 :	naabna	53 :	nbaaan

[입출] `permute.inp`의 첫 줄에 단어가 주어진다. 단어는 소문자로만 구성되어 있으며 그 길이는 최소 3, 최대 15이다. 그 다음 줄에는 k 값이 주어진다. $-k$ 번째 단어와 $+k$ 번째 단어를 각각 한 줄에 하나씩 2줄에 걸쳐 출력한다. 만일 해당 단어가 없으면 "none"을 대신 출력한다.

[예제]

<code>permute.inp</code>	<code>permute.out</code>	
banana	annbaa	28 : piazz
5	bnnaaa	29 : pizaz
		30 : pizza
pizza	piazz	31 : pzaiz
2	pzazi	32 : pzazi
		33 : pziaz

[조건] 프로그램의 이름은 `permute.cpp`이며 C언어(`permute.c`)도 가능하다. 제출횟수는 15회, 제한시간은 1초이다. 마감시간은 **11월 8일(토) 23시**, 제출은 11월 5일(목)부터 가능하다. 단 내장된 함수를 사용하면 안 되며(예를 들어 `next_permutation()`) 반드시 STL에서 제공하는 `stack <char>`, `stack<mytype>` 만을 사용해야 한다.