

# Suffix Array

**[문제]** 텍스트 문자열  $T$ 와 그에 대한 접미사배열(Suffix Array)과 최장공통접두사(LCP) 배열이 주어진다. 여러분은 이를 이용하여 패턴문자열 집합  $P=\{P_i\}$ 가 주어졌을 때 각각의 패턴문자열이  $T$  상에 몇 번 출현했는지를 출력하는 프로그램을 작성해야 한다.

**[입출력]** dna.txt 파일의 첫 줄에 문자열의 길이  $1 \leq N \leq 10,000$ 이 주어진다. 그 다음 줄에는 {a,g,c,t,\$}로 이루어진 텍스트 문자열  $T$ 가 주어진다 (단, 종료문자 \$는  $T$ 의 마지막 위치에만 주어진다.). 다음 줄에는  $N$ 개의 정수로 이루어진 접미사배열이 주어지고, 이어서  $N-1$ 개의 정수로 이루어진 LCP배열이 주어진다. 다음 줄에 질의문자열의 개수  $1 \leq M \leq 10,000$ 이 주어지고, 이어지는  $M$ 개의 줄에 질의문자열이 각 줄에 하나씩 주어진다. 질의문자열 각각의 길이는 100을 넘지 않는다.

이 때, 프로그래밍의 편의상 인덱스는 0부터 시작한다. 예를 들어  $T=abaab\$$ 이면 접미사배열은 다음과 같이 주어진다.

i	0	1	2	3	4	5
$T[i]$	a	b	a	a	b	\$
$SA[i]$	5	2	3	0	4	1
$LCP[i]$	0	1	2	0	1	

이 문자열에 대한 입출력 데이터 예시는 아래와 같다.

입력	출력
6	2
acaac\$	3
5 2 3 0 4 1	1
0 1 2 0 1	
3	
ac	
a	
ca	

**[조건]** 제출은 **6월 2일 화요일 저녁 10시**까지이다. 제출은 ESPA 과제물 게시판이며 제출물은 다음과 같다. **SufArray.{c,cpp,py}**에는 자신이 구현한 코드를, **NAME\_report.pdf**에는 실험결과 등을 기술한다.

보고서에는 1) 기본적인 방법(언어나 표준라이브러리에서 제공하는 `find()` 등)으로 구현한 경우와 2) 접미사배열을 이용해 단순한 이진 탐색한 경우 3) 접미사배열을 이용한 탐색 속도를 높이기 위한 테크닉을 추가적으로 적용한 경우에 대한 차이점이 기술되어야 한다.