

카테고리 없음

사전

Lisa_Ha 2023. 10. 10. 03:06 ⓘ

하...알고리즘 실습 과제로 멘탈이 털린 후의 알고리즘 스터디 아주 즐겁게 해볼게 ㅇㅍ.

문제

동호와 규완이는 212호에서 문자열에 대해 공부하고 있다. 김진영 조교는 동호와 규완이에게 특별 과제를 주었다. 특별 과제는 특별한 문자열로 이루어진 사전을 만드는 것이다. 사전에 수록되어 있는 모든 문자열은 N개의 "a"와 M개의 "z"로 이루어져 있다. 그리고 다른 문자는 없다. 사전에는 알파벳 순서대로 수록되어 있다.

규완이는 사전을 완성했지만, 동호는 사전을 완성하지 못했다. 동호는 자신의 과제를 끝내기 위해서 규완이의 사전을 몰래 참조하기로 했다. 동호는 규완이가 자리를 비운 사이에 몰래 사전을 보려고 하기 때문에, 문자열 하나만 찾을 여유밖에 없다. N과 M이 주어졌을 때, 규완이의 사전에서 K번째 문자열이 무엇인지 구하는 프로그램을 작성하시오.

입력

첫째 줄에 세 정수 N, M, K가 순서대로 주어진다.

출력

첫째 줄에 규완이의 사전에서 K번째 문자열을 출력한다. 만약 규완이의 사전에 수록되어 있는 문자열의 개수가 K보다 작으면 -1을 출력한다.

제한

- $1 \leq N, M \leq 100$
- $1 \leq K \leq 1,000,000,000$

예제 입력 1

2 2 2

예제 출력 1

azaz

예제 입력 2

2 2 6

예제 출력 2

zzaa

예제 입력 3

10 10 1000000000

예제 출력 3

-1

예제가 많네요 쉽지는 않겠어요.

요약하자면,
문자열로 이루어진 사전을 만드는 것
사전에 수록되어 있는 모든 문자열은 N 개의 "a"와 M 개의 "z"로 이루어져 있다.
그리고 다른 문자는 없다.
사전에는 알파벳 순서대로 수록되어 있다.
규완이는 사전을 완성했지만, 동호는 사전을 완성하지 못했다.
동호는 규완의 사전에서 문자열 하나만 찾는다
 N 과 M 이 주어졌을 때, 규완이의 사전에서 K 번째 문자열이 무엇인지 구하는 프로그램을 작성하시오.
뭐야 동호는 왜 나온거냐

=> 그렇다면 전체 문자열 수는 어떻게 될까? => $(N+M)C(N)$ 과 같지 않을까..?
=> $(N+M)! / (N! * M!)$
어우 너무 많다. 이 모든 경우의 수를 나열해서 k 번째를 구하는 방식은 아닐 것이다.

그렇다면.. 앞에 들어갈 a의 숫자를 c 라고 한다면
 $(N+M-c+1)C(N-c+1) \leq k \leq (N+M-c)C(N-c)$ 인 c 를 구한다면..? 경우의 수가 많이 줄어 든다.

즉
 c 개 만큼 a를 앞에 나열한 후 남은 $N-c$ 개의 a와 M 개의 z를 나열하는 경우를 찾자.

그렇다면 „하...머리아프다 코드로 하나하나 비교해 보면서 확인하자.
더보기

개수는 사전순으로 k 번째 이므로 앞에서부터 a개가 있다면, 남은 a는 num_a 개.
 $(M+\text{num_a})C(\text{num_a}) \leq k-1 \leq (M+\text{num_a}+1)C(\text{num_a}+1)$ 인 num_a 를 찾자.

그렇다면 num_a 까지 도달하기 전까지는 몇번째 일까?
 $\text{num_a}++$ 하기 전의 c 값이다.

더보기
즉, 0 개 라는 것이다.
 $k-0$ 개를 더 하면 되므로 num_a 개의 a와 M 개의 z를 나열하는 순서에서 $k-0$ 번째 문자열을 찾자..
아니 뭔가 너무 어렵게 가는 느낌..?

현재 배우는 알고리즘을 사용하지 않으므로 트리로 써야하는 건가..?

일단 마저 풀겠다.

이제 남은 알파벳을 나열 하면 되는데, 이를 트리로 할까 배열로 할까 고민하다가 배열을 선택했다. 트리 너무 어렵다.

num_a + K개를 나열 할 때
앞에서부터

	1	2	3	4	5
(1+2)-2=1	a	a	z	z	z
(1+3)-2=2	a	z	a	z	z
(1+4)-2 =3	a	z	z	a	a
(2+3)-1=4	z	a	a	z	z
(2+4)-1=5	z				
4					
5					
5					
6					
7					
7					
8					
9					

열의 숫자를 더하면 무엇인가 규칙이 나오는것 같아서 계속 고민을 해봤는데.. 모르겠다..

내 머리를 쥐어짜서 나오는 아이디어는 여기까지,...

아래는 작성된건 거의 없지만 작성한 코드이다..

더보기

혼자 고민한지도 3시간째 ... 이제 다른사람의 아이디어를 봐야겠다..

참고할 자료를 찾아보았다..

[백준 1256번 - 사전 \(tistory.com\)](https://www.acmicpc.net/problem/1256)

	<p>백준 1256번 - 사전</p> <p>https://www.acmicpc.net/problem/1256 1256번: 사전 동호와 규완이는 212호에서 문자열에 대해 공부하고 있다. 김진영 조교는 동호와 규완이에게 특별 과제를 주었다. 특별 과제는 특별한 문...</p> <p>barbera.tistory.com</p>
--	--

아니..필요한 배경지식이.

- 다이나믹 프로그래밍
- 조합론 - 파스칼 항등식

이러는데 파스칼은 들어봤어도 다이나믹 프로그래밍은 처음 들어봤다.

남의 코드를 그대로 작성해봤자 아무 도움이 안되니 다이나믹 프로그래밍이 무엇인지 알아보고 글을 마치겠다.

*다이나믹 프로그래밍, 한국말로 동적 계획법 알고리즘이라고 한다.

: 메모리 공간을 약간 더 사용해서 연산 속도를 비약적으로 증가시키는 방법

이 방법은 항상 쓸 수 있는건 아닌가 보다. 2가지의 조건이 붙는데.

1. 큰 문제를 작은 문제로 나눌 수 있다.
2. 작은 문제에서 구한 정답은 그것을 포함하는 큰 문제에서도 동일하다.

이것이 그 조건이다.

이게 무슨말인지 감이 하나도 안오는데,
해당 조건을 만족하는 것이 피보나치 수열이라고 한다.
만약 재귀식으로 푼다면 피노나치 수열은 연산이 길어지는데, 이때 연산시간을 줄이기 위해 재귀를 사용한, 동적계획법을 사용
한다고한다.
다이나믹 프로그래밍의 가장 큰 포인트는 바로 한 번 결과를 수행한 것을 메모리에 저장해 놓고 다음에 똑같은 결과가 필요하면
그 때 다시 연산하지 않고 메모리에 저장된 그 값을 가져와 쓰는 것이다.

우리가 아는 단순 재귀는 계속된 연산을 하는 것(러닝타임이 길다)이라면, 동적계획법(러닝타임이 짧다)과의 차이는 캐싱을 하느냐 안하느냐 인것 같다..
: 캐싱(메모제이션)

[Python] Dynamic Programming(동적계획법) 알고리즘 (tistory.com) - 참고자료

[Python] Dynamic Programming(동적계획법)...

이번 포스팅에는 최근에 Python으로 알고리즘을 공부하기 시작하면서 알게 된 여러 알고리즘의 원리와 Python으로 구현하는 방법에 대해 소개해보려 한다. 필자는 최근 알고리즘 공부를 '나동

techblog-history-younghunjo1.tistory.com

그래도 코드를 원한다면,, 나의 코드는 아니지만 첨부하겠다.
더보기

이건 나의 배경지식 이슈로,, 이해하지도 풀지도 못하는 문제이다..

공감

댓글 0

Lisa_Ha

내용을 입력하세요.

등록