

# 백준 9251

## LCS

Longest Common Subsequence

최장 공통 부분수열

# LCS란 무엇인가?

- LCS: Longest Common Subsequence  
공통 문자열이 연속되지는 않으나, 선후 관계가 일치하는 형태
  - ABCDEF
  - GBCDFE
- LCS: Longest Common Substring / 공통 문자열이 연속되는 형태
  - ABCDEF
  - GBCDFE

# LCS를 구하는 기본 방법

- LCS: Longest Common Substring
  - 2차원 배열을 이용해 접근한다.
- 문자가 같으면서 순서도 같아야함
- 두 문자가 같다면,  
Arr[i-1][j-1]의 값 + 1한 값을  
Arr[i][j]에 입력한다.

	0	A	B	C	D	E	F
0	0						
G		0	0	0	0	0	0
B		0	1	0	0	0	0
C		0	0	2	0	0	0
D		0	0	0	3	0	0
F		0	0	0	0	4	0
E		0	0	0	0	0	0

# LCS를 구하는 기본 방법

- LCS: Longest Common Subsequence

- 2차원 배열을 이용해 접근한다.

- 문자는 같으나 순서가 같지는 않다.
  - 그러므로, 현재의 문자를 비교하기 이전의 최대 공통 부분 수열은 유지가 된다.

- 두 문자가 다르다면,  
→  $Arr[i-1][j]$  혹은  $Arr[i][j-1]$  중 큰 값

- 두 문자가 같다면,  
→  $Arr[i-1][j-1]$ 의 값 + 1한 값을  
 $Arr[i][j]$ 에 입력

	0	A	B	C	D	E	F
0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	0
B	0	0	1	1	1	1	1
C	0	0	1	2	2	2	2
D	0	0	1	2	3	3	3
F	0	0	1	2	3	3	4
E	0	0	1	2	3	4	4

# LCS의 최대길이를 찾는 과정

- 앞서 채운 Arr에서의 마지막값을 찾으면 끝

	0	A	B	C	D	E	F
0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	0
B	0	0	1	1	1	1	1
C	0	0	1	2	2	2	2
D	0	0	1	2	3	3	3
F	0	0	1	2	3	3	4
E	0	0	1	2	3	4	4

# LCS를 찾는 과정

- 앞서 채운 Arr의 마지막 값부터 탐색한다.  
그 값과 같은 크기를 가진 배열을 우선 생성한다.

• result = [ , , , ]

- 1) Arr[i-1][j] or Arr[i][j-1] 중  
Arr[i][j]와 같은 값을 갖는 곳으로 이동
- 2) 그곳에서 1) 과정을 수행  
같은 값을 갖는 곳이 없다면,  
Arr[i][j]를 result에 저장한 후  
Arr[i-1][j-1]로 이동
- 3) 1) 2)의 반복

	0	A	B	C	D	E	F
0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	0
B	0	0	1	1	1	1	1
C	0	0	1	2	2	2	2
D	0	0	1	2	3	3	3
F	0	0	1	2	3	3	4
E	0	0	1	2	3	4	4

# Reference

- <https://abit.ly/oqbptx>