**동적 계획법 (Dynamic Programming)**

분할 정복(divide-and-conquer)과 유사.

작은 문제들의 답을 테이블에 저장해 놓았다 필요할 때 다시 계산하지 않고 테이블에서 가져와서 이용

1. 재귀적 속성을 찾고
2. 작은 문제들에 대한 답을 구해 테이블에 저장한 후
3. 필요시 테이블 내의 값을 이용한다.

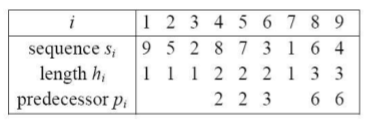
예시 ) 피보나치 수열

* F0 = 0
* F1 = 1
* Fn = Fn-1 + Fn-2 (n ≥ 2)
* N번째 피보나치 수를 구하는 문제는 N-1번째 피보나치 수를 구하는 문제와 N-2번째 피보나치 수를 구하는 작은 문제들로 구성될 수 있다.
* 이 때 F10을 구할 때 사용하는 F3의 값 = F9를 구할 때 사용하는 F3의 값 = … = F4을 구할 때 사용하는 F3의 값 인 것처럼 겹치는 부분문제의 합이 항상 같아야 한다.

예시 ) 가장 긴 증가하는 부분수열

S = (9, 5, 2, 8, 7, 3, 1, 6, 4) 이라는 수열의 경우 가장 긴 증가하는 부분 수열은

S = (9, 5, 2, 8, 7, 3, 1, 6, 4) 또는

S = (9, 5, 2, 8, 7, 3, 1, 6, 4) 이다.

hi = max (hj + 1) (0<=j<=i-1)

이러한 테이블로 확인할 수 있다.

예시) 숫자 삼각형

가장 아래에서부터 위로 올라가면서 숫자의 합이 최소가 되는 경로 및 그 최소값 구하기

