



동적 계획법(Dynamic Programming)

정의

- 하나의 복잡한 문제를 여러 개의 간단한 문제로 나누어 푸는 방법
 - 부분 문제를 통해 큰 문제를 재귀적으로 해결하려고 할 때 중복되는 부분 문제들이 많이 발생하여 불필요하게 같은 계산이 반복될 때 사용
- 주어진 문제를 풀기 위해서, 문제를 여러 개의 하위 문제로 나누어 푼 다음, 그것을 결합하여 최종적인 목적에 도달한다.
- 하위 문제의 수가 기하급수적으로 증가할 때 유용!
- 최단 경로 문제 행렬의 제곱 문제 등의 최적화에 사용됨
- Memoization 기법 사용
 - Memoization: 프로그램 실행 시 이전에 계산한 값을 저장하여, 그 값이 필요할 때 그 값을 참조할 수 있도록 하는 방식

구현 방식

1. Top-down

1. 큰 문제를 작은 문제로 쪼갬다.
2. 작은 문제를 푼다.
3. 작은 문제를 풀었으니 이제 큰 문제를 푼다.

재귀로 구현

2. Bottom-up

1. 크기가 작은 문제부터 차례대로 푼다.
2. 문제의 크기를 조금씩 키워가며 푼다.
3. 반복하다 보면 가장 큰 문제를 풀 수 있다.

반복문으로 구현

DP로 문제풀이하는 방법

1. 먼저 문제에서 구하려고 하는 답을 문장으로 나타낸다.
2. 그 문장에 나와있는 변수의 개수만큼 메모하는 캐시 배열을 만든다.
 - a. Top-down인 경우에는 재귀 호출의 인자의 개수가 된다.
3. 마지막으로, 문제를 작은 단위로 쪼개 뒤 수식(점화식)을 이용해서 문제를 표현해야 한다.