

## 백트래킹

기본적으로 백트래킹은 '가능한 모든 방법을 탐색한다' 는데 기본 아이디어가 있다.

대표적인 완전 탐색 방법으로는 **DFS (Depth First Search, 깊이 우선 탐색)** 이 있다. **DFS** 는 현재 지점에서 방문할 곳이 있으면 재귀 호출을 이용해서 계속 이동한다.

**DFS** 의 장점은 무한히 깊은 곳을 찾아야 할 때 효과적이다. 하지만 **DFS** 는 모든 곳을 방문하기 때문에 굳이 목표지점이 있지 않는 경로로 빠져서 비효율적인 결과를 초래할 수도 있다.

그래서 이와 같은 비효율적인 경로를 차단하고 목표지점에 갈 수 있는 가능성이 있는 루트를 검사하는 방법이 백트래킹 알고리즘이다. 백트래킹은 **DFS** 에 가지치기 (**Pruning**) 를 통해 가도되지 않는 루트는 고려하지 않고 탐색하는 완전탐색 기법이다. 가지치기를 얼마나 잘하느냐에 따라서 효율이 극대화 될 수 있는 방법이다.

**DFS** 는 가능한 모든 경로(후보)를 탐색합니다. 따라서, 불필요할 것 같은 경로를 사전에 차단하거나 하는 등의 행동이 없으므로 경우의 수를 줄이지 못합니다. 따라서  $N!$  가지의 경우의 수를 가진 문제는 **DFS** 로 처리가 불가능할 것입니다.

**백트래킹이란 DFS 를 사용하고 가지치기(=불필요한 반복문을 줄이는 것)**

**즉, 백트래킹이란 특정한 경우만 살펴보는 것**

주로 문제 풀이에서는 **DFS** 등으로 모든 경우의 수를 탐색하는 과정에서, 조건문 등을 걸어 답이 절대로 될 수 없는 상황을 정의하고,

그러한 상황일 경우에는 탐색을 중지시킨 뒤 그 이전으로 돌아가서 다시 다른 경우를 탐색하게끔 구현할 수 있습니다.