

그래프는 노드와 간선으로 구성된 자료구조이다. 노드는 데이터를 저장하는 공간이고 간선은 노드 간의 관계를 나타낸다.

1. 노드 : 데이터를 저장하는 기본 단위이다.
2. 간선 : 두 노드를 연결하는 선이다. 방향성이 있을 수도 있고 없을 수도 있다.
3. 인접 : 간선으로 연결된 두 노드를 서로 인접하다고 한다.

그래프의 종류

1. 무방향 그래프 : 간선에 방향성이 없는 그래프이다. A에서 B로 갈 수 있다면 B에서 A로도 갈 수 도있다.
2. 방향 그래프 : 간선에 방향성이 있는 그래프이다. A에서 B로 갈 수 있다고 해서 B에서 A로 갈 수 있는 것은 아니다.
3. 가중치 그래프 : 간선에 가중치가 부여된 그래프이다. 이 가중치는 거리, 비용 등을 나타낼 수 있다.



인접 리스트

그래프를 표현하는 방법 중 하나이다. 각 노드에 연결된 인접 노드들을 리스트 형태로 저장한다.

1. 장점 : 간선의 수가 적은 희소 그래프에서 효율적이다. 메모리 공간을 절약할 수 있다.
2. 단점 : 모든 노드 간의 연결 여부를 확인하려면 모든 리스트를 순회해야 한다.



깊이 우선 탐색(Depth-First Search, DFS)

그래프 탐색 알고리즘 중 하나로 하나의 가지를 끝까지 탐색한 후 다른 가지를 탐색하는 방식이다. 스택 자료구조를 이용하여 구현하며 미로 탐색, 사이클 검출 등에 사용된다.



너비 우선 탐색(Breadth-First Search, BFS)

그래프 탐색 알고리즘 중 하나로 현재 노드와 인접한 모든 노드를 먼저 방문하고 그 다음에 그 노드들과 인접한 노드들을 방문하는 방식이다. 큐 자료구조를 이용하여 구현하며 최단 경로 찾기, 연결 요소 찾기 등에 사용된다.



DFS와 BFS의 차이점

특징	DFS	BFS
탐색 방식	깊이 우선	너비 우선
자료 구조	스택	큐
주요 용도	미로 탐색, 사이클 검출	최단 경로 찾기, 연결 요소 찾기

비선형 자료구조 그래프의 활용

1. 소셜 네트워크 분석 : 친구 관계, 추천 시스템 등을 구현하는 데 사용된다.
2. 최단 경로 찾기 : 길찾기 앱, 네트워크 라우팅 등에 활용된다.
3. 파일 시스템 : 파일과 디렉토리의 관계를 트리 형태의 그래프로 표현할 수 있다.

