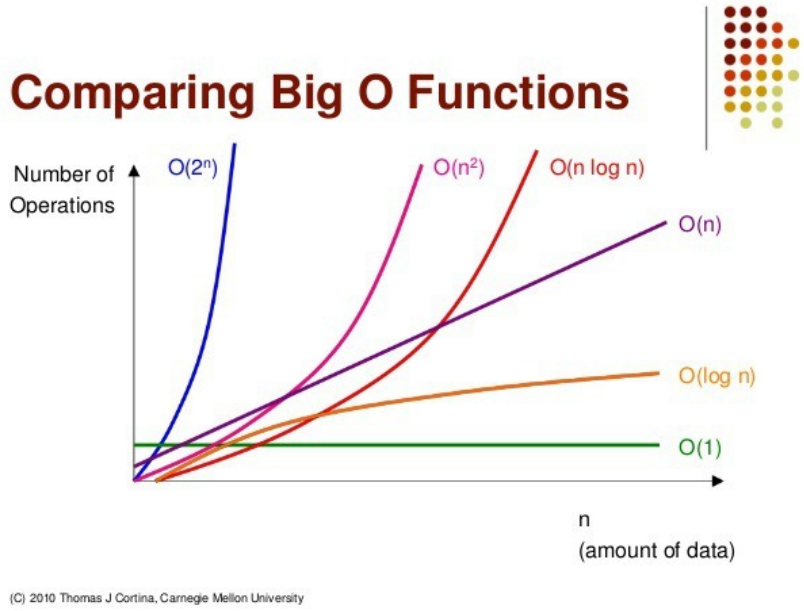
# 12주차

- **빅오 표기법에 대해서 설명해주세요**

알고리즘 효율성을 표기하는 방식
빅오: 최악의 실행 시간, 가장 많이 사용
빅오 표기법의 특징
-상수항 무시
-계수 무시
-최고차항만 표시




O(N log N) 퀵 / 병합 / 힙 정렬
O(N^2)  삽입/버블선택 정렬
O(2^N) fibonacci 수열

+)
빅오메가: 최상의 실행 시간
빅세타: 평균 실행시간

- **피보나치 수열 구현 방식 세 가지를 말해보시고, 시간복잡도와 공간복잡도를 설명해 주세요.**

F(n) = F(n-1) + F(n-2) (n ≥ 2)
for문, 재귀, dp

-for문
시간복잡도 O(n)
공간복잡도: O(n) 배열

-재귀함수
시간복잡도는 O(2^n)
공간복잡도: O(n) 재귀 호출 스택 깊이
이미 계산된 값을 다시 계산하여

```
return fibo(x - 1) + fibo(x - 2);
```

-dp
시간복잡도 O(n)
공간복잡도: O(n) 메모제이션 값 저장
메모이제이션(memoization)을 활용해 이전에 계산해둔 값을 재사용
+)tail recursion

- **BFS/DFS 차이는 무엇인가요?**

두가지 방법 모두 그래프 탐색시 사용

-BFS
너비우선탐색
현재 정점에서 가까운 정점들부터 탐색
큐를 이용해 구현
최단거리 구할 때 사용
O(V + E)
위상 정렬시 사용 (커리큘럼같은 거)

-DFS
깊이우선탐색
현재 정점에서 깊이 들어갈 수 있는 정점부터 탐색
재귀함수로 구현
경로의 특징을 고려해야할 때 사용
O(V + E)
사이클 탐색할 때

- **프림 알고리즘에 대해서 설명해 주세요.**

MST(최소신장트리) 구현시 사용되는 알고리즘
신장트리:최소연결그래프
최소신장트리: 신장트리인데 연결된 간선들의 가중치 합이 최소인 트리
그리디 기반
탐색한 정점들의 가까운 정점들 중 최소 비용의 간선으로 연결된 정점을 선택
이때, 사이클이 생기면 안됨
인접 정점들 중 최소 비용 간선의 정점을 찾기 위해 우선순위 큐를 사용
인접 정점들을 순회하며 우선순위 큐에 삽입한 후 pop하여 사용
O(E log V)

+)신장 트리=스패닝 트리
+)크루스칼 알고리즘으로 MST 구현 더 많이 함

- **다익스트라 알고리즘에 대해서 설명해 주세요.**

dp를 이용한 최단 경로 탐색 알고리즘
특정한 하나의 정점에서 다른 모든 정점으로 가는 최단 경로
최단 거리는 여러개의 최단거리로 이루어져있음을 이용하는 알고리즘
주변 정점 중 최소 비용 정점으로 이동하며 이동한 해당 정점에서 다음 정점으로 이동시 비용이 감소한다면 해당 비용으로 값을 업데이트, 이후 다시 주변 최소 비용 정점으로 이동하여 로직 반복
배열 이용 시간복잡도: O(V^2)
우선순위 큐 이용 시간복잡도 O(ElogV)
+)네비게이션에서 쓰이는 알고리즘

- **은행원 알고리즘에 대해서 설명해 주세요. → 다음주**

데드락 회피 알고리즘
자원의 할당 여부를 결정할 때 미리 결정된 모든 자원의 최대 가능 할당량을 고려해
안전한 상태가 보장될 때에만 할당
스레드가 자원을 요청하면 해당 자원을 수용했을 때 안전 상태 유지가 가능한지 파악 후 안전 상태일때만 할당

단점
-최대 자원 요구량을 미리 계산해 알고 있어야함
-항상 불안정 상태를 회피해야하므로 자원 활용도가 낮음

```
int gcd(int n, int m) {
if (n%m ==0) return m;
if (n < m) swap(n, m);
while (m > 0) {
n = n%m;
swap(n, m);
}
return n;
}
```

답: logn

1. `int count = 0;`
   `for (int i = N; i > 0; i /= 2) { for (int j = 0; j < i; j++) { count += 1; }`
   `}`

답: n