

2차원리스트 탐색

인접리스트

BFS

너비 우선 탐색

큐, 자료구조 구현

deque

그래프 탐색

DFS

귀환점 탐색

스택, 재귀함수로 구현

append pop

인접행렬

연산비용도  
복잡비용  
1eq

비크리만  
대거의 사용  
가능함

삽입정렬

정렬

선택정렬

계수정렬

퀵정렬

실용적인  
최대 가능

최소화할  
배열의 첫 데이터를  
피벗으로 사용

데이터의 가장 작은 값부터  
가장 큰 값까지의  
배열 생성

0으로 초기화

정렬할 데이터의  
인덱스 위치는  
1 대입

$O(N+K)$

마무 빠르다  
가장 큰 데이터의  
가장 작은 데이터의 차이가  
1,000,000 이하인  
때에 효용적

$O(N^2)$

최선의 경우 (정렬된 경우)  
 $O(N)$

다른 값이 작으면  
원점으로 더 작은 값을  
만날 때까지 이동

만나면  
그 다음 위치에서  
삽입

첫번째 값에서  
작은 값을 시작

데이터가  
순차적일수록  
빠르다

가장 작은 데이터를  
맨 앞으로

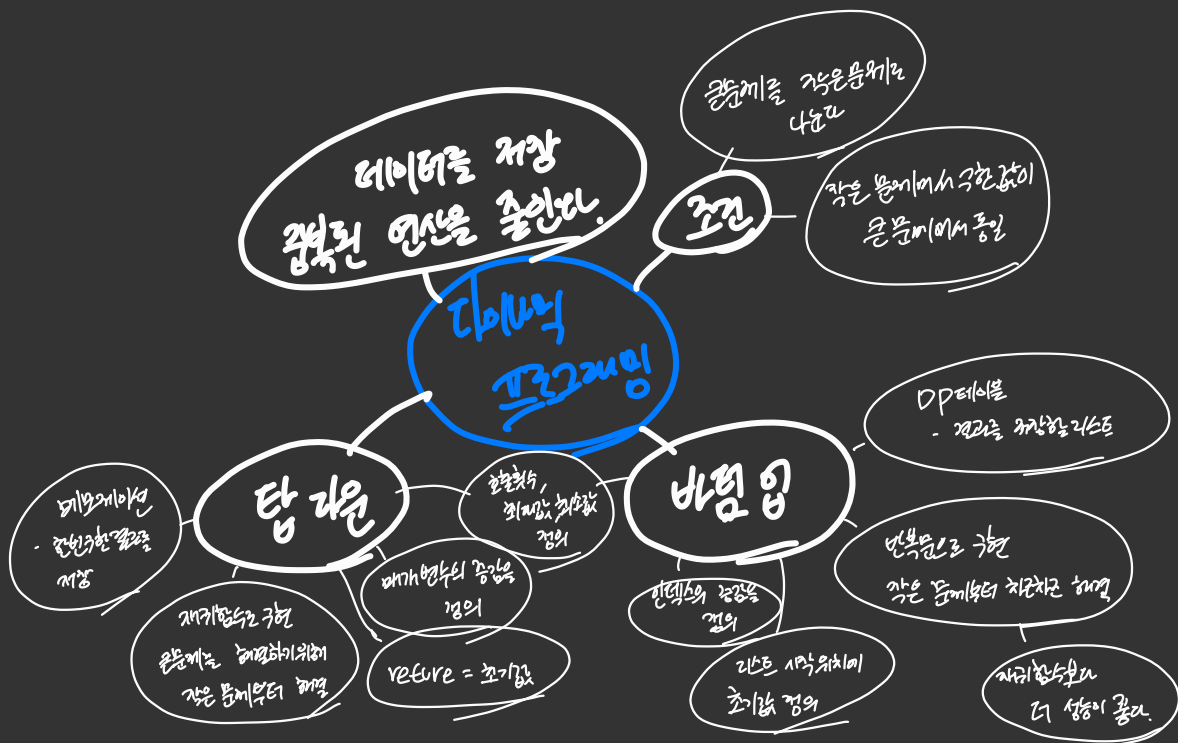
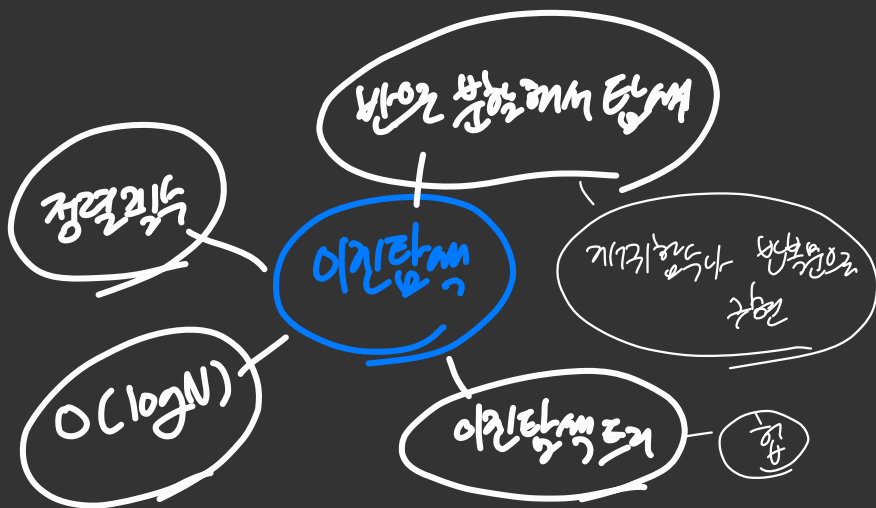
정렬된 위치를 빼고  
다시  $N-1$ 번 정렬

$O(N^2)$

최악의 경우 (피벗이  
가장 큰 값)  
 $O(N^2)$

$O(N \log N)$

피벗을 정렬된  
오른쪽, 왼쪽에서  
각각 탐색  
오른쪽은 피벗보다 작은 값  
왼쪽은 피벗보다 큰 값에서  
발견하고 교환  
탐색이 교차하면  
작은 값과 피벗 교환  
피벗값으로 분할해서  
반복





## 72차 이론

