선형 검색(linear search)

순차적으로 검색

데이터 100개 일 때

worst case : 100

best case : 1

average case : 50.5

이진 검색(bineary search)

데이터를 계속 반으로 쪼개서 검색, 단 정렬된 데이터만 가능하다

데이터 100개 일 때

worst case : 100

best case : 1

average case : 50.5

버블 정렬(bubble sort)

시작점에서 끝까지 가면서 제일 높은 거 맨 끝으로 이것을 계속 반복

선택 정렬 (select sort)

시작점에서 한바퀴 돌아 제일 작은 것을 놓는다

삽입 정렬 (insert sort)

시작점 기준 오른쪽과 비교 오른쪽이 더 작으면 교체. ex)초등학교 학생들 키 순으로 배열할 때 하는 거

퀵 소트 (quick sort)

기준점으로 작은 값은 앞으로 큰 값은 뒤로 보내는 정렬. 재귀함수 이용함

정리사항

1) DWORD : UNSIGNED LONG 자료형, windows api에서 사용

2) memset 함수 : 어떤 메모리의 시작점부터 연속된 범위를 어떤 값으로(바이트 단위) 모두 지정하고 싶을 때 사용하는 함수

-> 함수 구조

void\* memset(void\* ptr, int value, size\_t num);

- ptr : 채우고자 하는 메모리의 시작 포인터(시작 주소)

- value : 메모리에 채우고자 하는 값. Int 형이지만 내부에서는 unsigned char (1byte)로 변환되어서 저장된다.

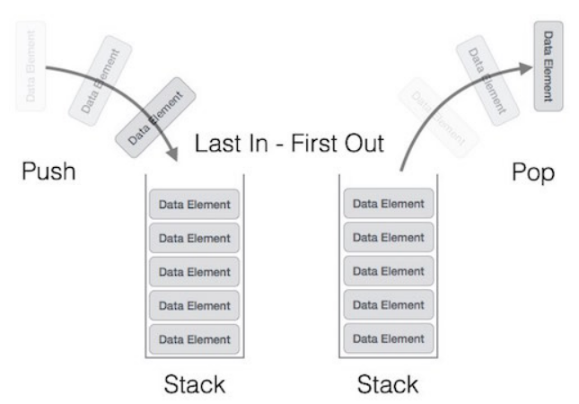
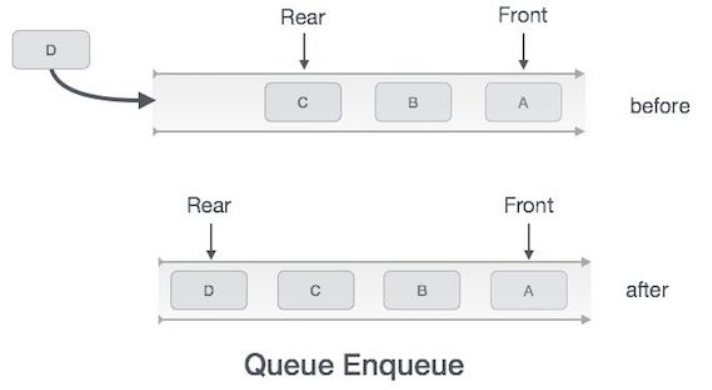
- num : 채우고자 하는 바이트의 수. 즉 채우고자 하는 메모리의 크기

3) TCHAR : windows에서 자료형이 바꿔진 거 -> char 또는 wchar

4) generic programming : 어느 타입의 자료형이든 간에 특정 기능을 수행하기 위해 한 함수만 호출해도 가능하게 작성하는 기법

5) Stack 스택 : 데이터가 입력되면 입력되는 순서대로 쌓고, 나중에 들어온 것부터 먼저 사용하는 자료구조 (선입후출)

6) 큐 (queue) : 먼저 넣은 데이터가 먼저 나오는 선입선출 형식



7) 동적할당 : 컴퓨터 프로그래밍 하는 중 실행 시간 동안 사용할 메모리 공간을 할당 하는 것

배열 vs 리스트

* 배열 : 여러 데이터를 하나의 이름으로 묶어서 관리 하기 위한 자료구조. 배열은 정의하면 길이를 지정하고 길이를 바꿀 수 없다. 그래서 길이보다 적게 사용하더라도 기존 용량은 변화가 없고 쓸데 없는 용량이 생기게 된다.
* 리스트 : 빈 공간 없이 데이터를 적재하기 위해 만든 자료구조. 배열과 같은 인덱스는 없고 몇 번째 데이터인지의 의미를 가지고 있다. 데이터 개수에 따라 용량이 움직인다.

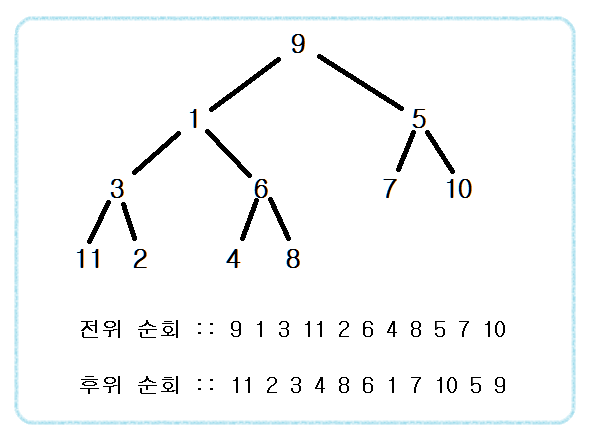
이진 트리 (binary tree) : 최대 2개의 자식 노드를 가진, 계층 구조를 가진 자료구조

* 차수 : 특정 노드의 자식 노드의 수
* 높이 : 루트 노드에서 단말 노드까지의 깊이
* 레벨 : 각 층의 번호 ex)루트 노드는 레벨 1이다.

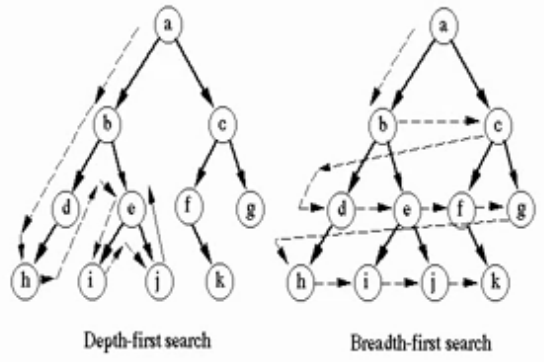
노드

값

전위 순회 & 후위 순회



깊이 우선 탐색 & 너비 우선 탐색



참고 자료

1. <http://soen.kr/> -> winAPI 참고함
2. 도서 Playing with C
3. 도서 do it c로 배우는 알고리즘
4. 도서 컴퓨팅 사고와 창의적 코딩
5. 도서 C premier plus
6. 기타 구글링 및 블로그