알고리즘 정리

==========

알고리즘 - 어떤 문제를 풀기위한 절차 공식적으로 표현

정렬 - 가장 느린 것(버블정렬), 가장 빠른 것(퀵정렬)
버블정렬, 선택정렬, 퀵정렬

빅오표기법 - 성능표현법, 가장 최악일 때 기준 표현!
a: 10, 60, 120
b: 5, 100, 400

String - length(), length(X)
equals(): 값을 비교할 때
s = "hong", s2 = "kim"
s.substring(1) //ong
s.indexOf("h") //0
클래스, String s3 = "aaa"; s3 ="bbb";
String은 값이 변경되면 새로운 공간을 만든다.(원래의 저장공간을 사용하지 않는다.)
자주 변경되는 경우, String을 쓰면 비효율적이다.
String은 값의 변경이 적은 경우 사용하면 효율적

순서도 - 직사각형(처리), 다이아몬드(조건, 분기-true/false), 종이찢어진 사각형(출력)

stack - 값을 저장공간에 넣는 것(push), 꺼내는 것(pop)
맨 위(top), 맨 아래(bottom)
push하면, top이 하나 증가한다.
top은 윗부분을 나타낸다.
꺼낼때는 top에서 먼저 꺼낸다.
일주일치 책을 쭉 쌓아놓고, 위에서 부터 읽어서 없앤다.

정렬 알고리즘 - 오름차순(1, 2, 3, 4), 내림차순(4, 3, 2, 1)
버블, 선택, 퀵 빅오로 표현 최악의 상황 ==> O()

퀵 정렬 - 기준값(피벗) 1, 3, 9, 7, 5 ==> 1, 3, 5(피벗), 7, 9
피벗값은 중간값을 사용함.

ArrayList<String> - String이외에는 넣을 수 없고, ArrayList - 기본타입은 Object을 설정
String데이터가 들어간 순서있는 데이터의 모음

시간복잡도는 cpu의 처리시간, 공간복잡도는 ram의 사용량을 말한다.

자료구조 - 중복X, 순서가 없는 데이터 구조(set)

큐(대기줄) - FIFO(선입선출구조, 먼저들어온 것이 먼저 나간다.),
큐의 앞부분(front), 뒷부분(back, rear)
큐의 삭제는 front에서 이루어진다.

List - FIFO(X), FILO(X), LinkedList, ArrayList, 순서O, 인덱스로 접근

HashMap - 키 + 값, JSON, 키는 유일O/중복X, 값은 유일X