Day 01
[자료구조 알고리즘 핵심 요점 정리]

강사님 메일:
yoonkwanghee1225@naver.com

중요한건 한가지를 배우고 넘어가야함
알고리즘 책 필요하면 이야기하세요*
우리나라 문제는 미국, 외국 문제 번형문제가 많이 나옴.
우리나라문제가 조금 더 쉬움

자료구조 (데이터의 구조) :

- 특정한 데이터를 찾기 위해서 어떤 방법을 구현하는것
- 데이터를 검색하는 논리

1. 선형구조 (linear structure)
    a. List
    b. Stack
    c. Queue
2. 비선형 구조 (non-linear structure)
    a. Tree
        i. stack
        ii. queue
    b. Graph
    c. heap
3. 기타 (others)
    a. sort
    b. back-tracking
    c. Implementation (구현)

알고리즘 (자료구조 데이터를 효율적으로 검색하는 방법):

1. Greed (크루스칼, 프림)
2. 다익스트라 (벨만 포드)
3. BSF / DFS (큐, 스택)
4. 구현 (현재 전체 비율에 36%)

2번째는 그리드 알고리즘의 기본문제임. 중요함

Collection (데이터의 집합):

- 여러 가지의 값을 담는 데이터 타입(Data type)을 이야기한다.
- 파이썬은 자동으로 데이터 타입을 바꿔준다 (파이썬이랑 다른 언어의 차이점).
    - 다른 언어(JAVA, C++, etc.)
        - *int num = 10;*
        - *num = 10.2; X (데이터 타입을 지정 해야한다)*
    - 파이썬
        - *num = 10*
        - *num = 10.2 O (데이터 타입을 지정 안해도 된다 -자*동형 변환환)
- 대표적으로, 리스트, 튜플, 딕셔너리, 셋, 문자열… 등이 있다.
- 컬렉션들은 데이터의 수정 가능여부에 따라 **변경할 수 있는 객체** 와 **변경할 수 없는 객체**로 나누어서 이야기 한다.
    - **뮤터블 객체 (변경할 수 있는 객체 - 수정가능)**
        - 변경할 수 있는 객체
        - 리스트(List), 딕셔너리(dict), 셋(Set)
            - List (two kinds)
                1. array list
                2. linked list
    - **이뮤터블 객체_자료구조 (변경할 수 없는 객체)**
        - 튜플(tuple)

- **파이썬의 기본은 결과값을 문자로 바꿔준다*****
    - **다른 언어에서는 데이터 타입을 구분해줘야 한다.**
    - **Input() 함수도 문자열로 바꿔준다.**
    - ex)
        - *“A” ⇒ ‘A’ (output)*
        - *“AB” ⇒ ‘AB’ (output)*
        - *23213 ⇒ ‘23213’ (output)*

**모든 코딩테스트의 값은 배열(array)로 나타내게 되어있다 (뮤터블 객체).

Stack

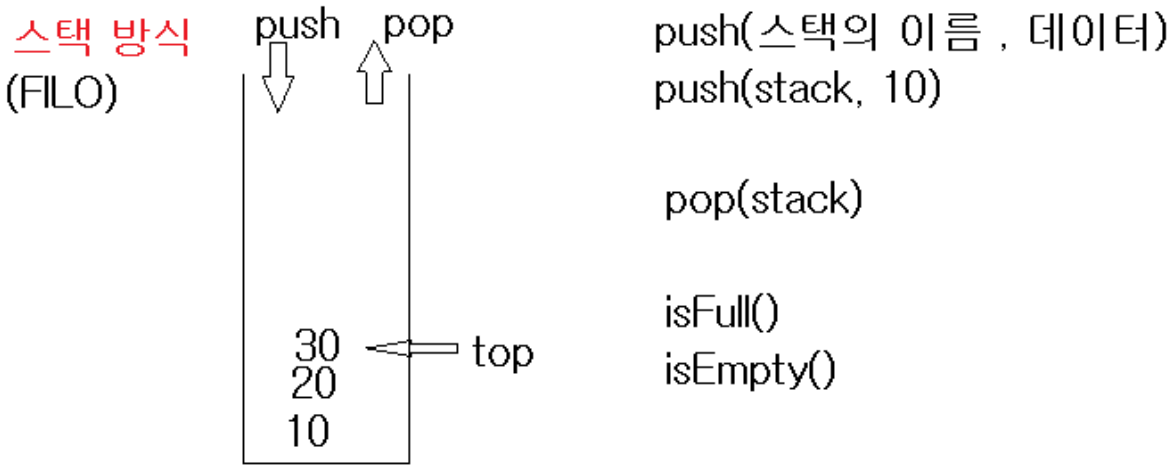
완전 탐색(Brute Force Search)

- 대표적인 알고리즘
    1. **DFS (Depth First Search)**
        - 깊이 우선 탐색
        - 탐색하고자하는 데이터의 검색을 스택(Stack) 방식을 쓴다
    2. **BFS (Breadth First Search)**
        - 너비 우선 탐색
        - 탐색하고자하는 데이터의 검색을 큐(Queue) 방식을 쓴다

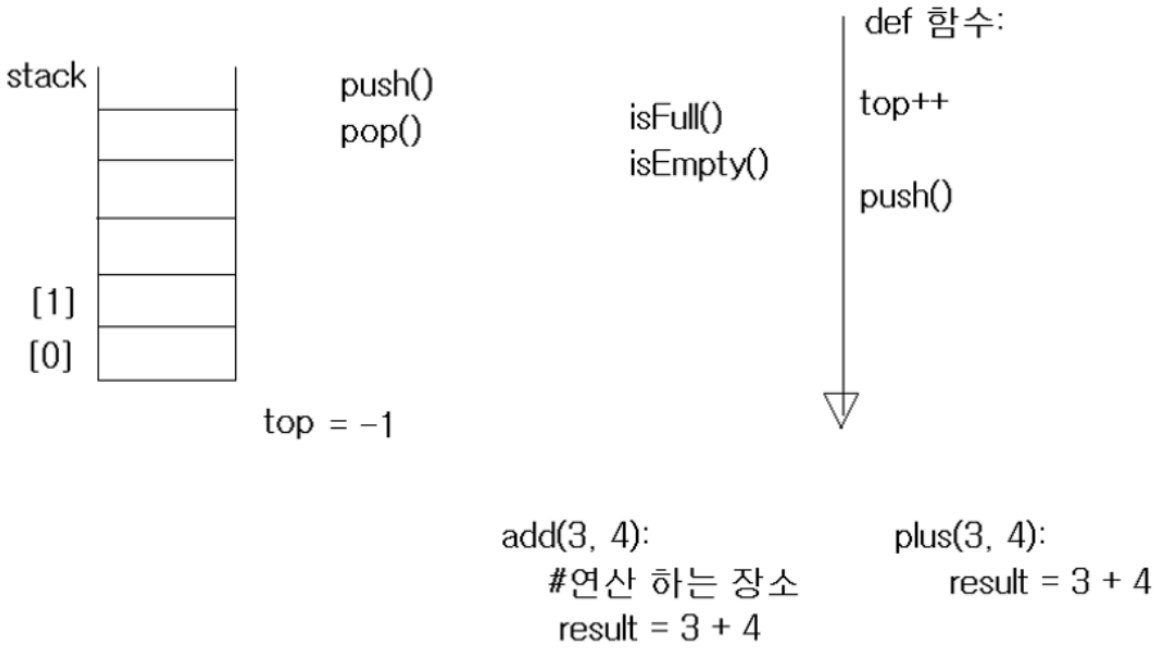
스택 방식 (FILO)

- **First In Last Out**
- **데이터를 넣을때 쓰는 함수**
    - insert, input, push, add, append, pullin, include
    - 다 사용가능하나, 추상적으로 사용할 수 있다 (ADT - Abstract Data Type).
        - 사용자가 원하는대로 사용 할 수 있다.
    - **자료구조는 기본적으로 push를 사용한다 (데이터를 집어넣는다는 뜻)**
        - push(스택의 이름, 데이터)

          ex) push(stack, 10)
- **데이터를 빼내는 함수**
    - pop, pick
        - **pop: 데이터를 빼내는것**
            - pop(stack)
            - 빼낼 데이터를 지정 안해도 맨위부터 뺀다
        - pick: 복사해서 데이터를 빼내는것
- **데이터가 들어갈 자리가 있는지 확인하는 단어**
    - isFull()
    - isEmpty()
- **현재 데이터가 몇번째 방에 있는지 알려주는 명령어**
    - top()
    - 데이터가 들어오기 전에 먼저 올라가서 확인한다.



1. 제일 먼저 비어있는 스택을 생성한다.
    a. stack = []
2. push() 함수에 의해 데이터를 삽입한다.
    a. push(stack, 10)
3. 2번을 실행하기전에 먼저 stack에 데이터를 삽입할 수 있는지 검사한다.



add(3, 4):
    #연산 하는 장소
    result = 3 + 4

plus(3, 4):
    result = 3 + 4