알고리즘 특강
정렬

자료구조에서 정렬을 공부하셨다면 알겠지만, 정말 중요한 내용입니다.
이번 시간엔 언어에서의 정렬을 주로 배우며, 활용은 다른 부분에서 볼 수 있습니다.

빠르게 복습하자.

Sorting

- FIFO (First In, First Out) 구조를 띄고 있는 자료구조로, 삽입과 삭제 연산이 서로 다른 한군데에서 발생함.
- 느린 알고리즘의 경우 시간복잡도가 $O(N^2)$, 빠른 경우 O(NlogN) 정도 된다.

Python에서의 정렬

```
_list = [4, 2, 1, 3, 1]

sorted_list = sorted(_list)
print(' '.join(map(str, sorted_list)))

_list.sort()
print(' '.join(map(str, sorted_list)))

wanna_to_eat = [
    ('Chicken', 17900, 'Puradak'),
    ('Pizza', 21000, 'Domino'),
    ('Spagetti', 12000, 'Mola')
]

wanna_to_eat = sorted(wanna_to_eat, key=lambda x: x[1])

wanna_to_eat = sorted(wanna_to_eat, key=lambda x: x[1], reverse = True)

_list1 = [21, 61, 4, 31, 65, 98]
_list2 = [66, 12, 34, 58, 91, 3]
_dict = dict(zip(_list1, _list2))
sorted_dict = sorted(_dict.items())

for key, item in sorted_dict:
    print("dictionary[{}] = {}".format(key, item))
```

- 자료형에 대해 **오름차순**으로 정렬을 하고, 결과값을 리턴함.
- 리스트의 메소드로, 내부 정렬을 함. (리스트를 제외한 **다른 자료형에선 불가!**)
- **Key** 옵션을 통해 정렬 기준을 지정함.
- 내림차순 정렬을 하기 위해 **reverse** 사용.
- Sorted를 딕셔너리에 사용 시 **키만 리턴 됨.** 따라서, 전체 조합을 유지하려면 items 사용.

Lambda

```
def adder(x, y):
    return x + y

adder(10, 20)
```

Lambda

```
(lambda x, y: x + y)(10, 20)
```

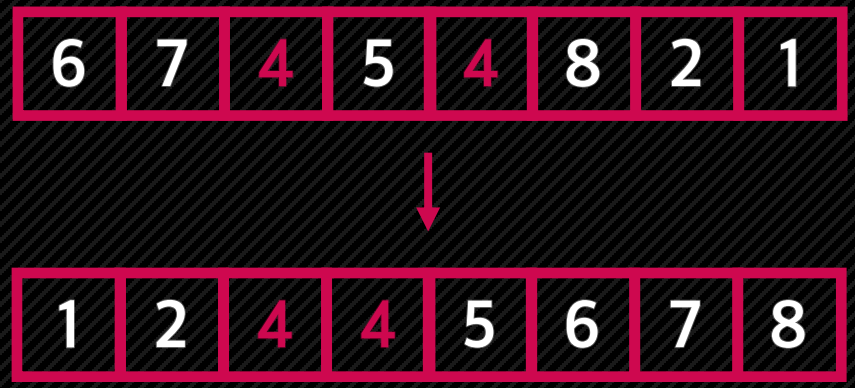
Lambda

- 익명함수로, 특정한 부분에서 임시적으로 생성해서 사용하고, 이후 버릴 수 있다.
- sort, map, filter, reduce 같은 함수/메소드에서 사용될 수 있다.

Sort에서의 Lambda

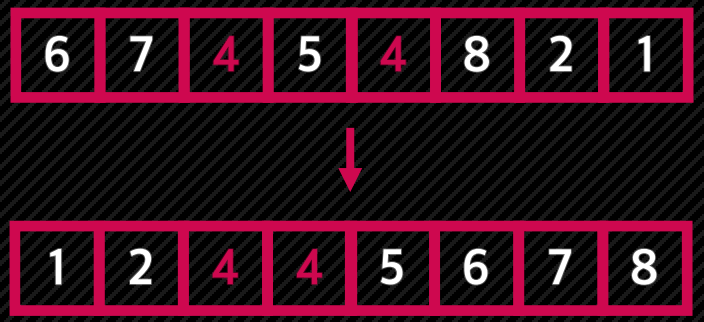
```
lotteria = [
    ('Hanwoo-Bulgogi', 7200, 8100),
    ('AZ', 6800, 7700),
    ('DoubleX2', 5700, 6900),
    ('Hot-Crispy', 4900, 6100),
    ('Miracle', 5700, 5600),
    ('Teri', 2700, 4000)
]

print(sorted(lotteria))
print(sorted(lotteria, key = lambda x: x[1]))
print(sorted(lotteria, key = lambda x: x[2] - x[1]))
print(sorted(lotteria, key = lambda x: (x[2] - x[1], x[0])))
```

Stable Sort



- 동일한 값의 데이터가 있을 때, 순서가 바뀌지 않음이 보장되는가?

Python은…?



- 동일한 값의 데이터가 있을 때, 순서가 바뀌지 않음이 보장되는가?

- Python의 정렬 알고리즘은 Timsort로, Stable함이 보장됨!

딱히 연습할게 없네요…

Silver 5 - 나이순 정렬 (#10814)

요약

- 온라인 저지에 가입한 사람들의 나이와 이름이 가입한 순서대로 주어진다.
- 나이가 증가하는 순으로, 나이가 같으면 먼저 가입한 사람이 앞에 오는 순서로 정렬하는 프로그램을 작성하시오.

제약조건

- 총 인원의 범위는 1 <= N <= 100,000 이다.

Silver 5 - 좌표 정렬하기 2 (#11651)

요약

- 2차원 평면 위의 점 N개가 주어진다.
- 좌표를 y좌표가 증가하는 순으로, y좌표가 같으면 x좌표가 증가하는 순서로 정렬하는 프로그램을 작성하시오.

제약조건

- 점의 수는 1 <= N <= 100,000 이다.

Level 2 - 파일명 정렬 (kakao blind recruitment 2018 3차 코딩테스트)

요약

- 파일명을 HEAD, NUMBER, TAIL의 세 부분으로 구성하여 정렬하려고 한다.
- HEAD는 숫자가 아닌 문자로 이루어져 있으며, 최소한 한 글자 이상이다.
- NUMBER는 한 글자에서 최대 다섯 글자 사이의 연속된 숫자로 이루어져 있으며, 앞쪽에 0이 올 수 있다.
- TAIL은 나머지 부분으로, 숫자가 다시 나올 수도 있고, 문자가 아예 없을 수도 있다.
- HEAD (대소문자 구분 없이) - NUMBER (숫자 순)으로 정렬하되, 만약 두 값이 모두 같으면 주어진 순서를 유지한다.

제약조건

- 전체 파일 수는 1,000개 이하이다.
- 각각의 파일은 100 글자 이하로, 영문, 숫자, 공백, 마침표, -로 이루어져 있다.

“Any Question?”