



최소직사각형

📅 날짜	@2024년 1월 16일
🏷️ 태그	Python 브루트포스 정민 프로그래머스

📖 문제

명함 지갑을 만드는 회사에서 지갑의 크기를 정하려고 합니다. 다양한 모양과 크기의 명함들을 모두 수납할 수 있으면서, 작아서 들고 다니기 편한 지갑을 만들어야 합니다. 이러한 요건을 만족하는 지갑을 만들기 위해 디자인팀은 모든 명함의 가로 길이와 세로 길이를 조사했습니다.

아래 표는 4가지 명함의 가로 길이와 세로 길이를 나타냅니다.

명함 번호	가로 길이	세로 길이
1	60	50
2	30	70
3	60	30
4	80	40

가장 긴 가로 길이와 세로 길이가 각각 80, 70이기 때문에 80(가로) x 70(세로) 크기의 지갑을 만들면 모든 명함들을 수납할 수 있습니다. 하지만 2번 명함을 가로로 눕혀 수납한다면 80(가로) x 50(세로) 크기의 지갑으로 모든 명함들을 수납할 수 있습니다. 이때의 지갑 크기는 4000(=80 x 50)입니다.

모든 명함의 가로 길이와 세로 길이를 나타내는 2차원 배열 `sizes`가 매개변수로 주어집니다. 모든 명함을 수납할 수 있는 가장 작은 지갑을 만들 때, 지갑의 크기를 `return` 하도록 `solution` 함수를 완성해주세요.

제한사항

- `sizes`의 길이는 1 이상 10,000 이하입니다.
 - `sizes`의 원소는 `[w, h]` 형식입니다.
 - `w`는 명함의 가로 길이를 나타냅니다.
 - `h`는 명함의 세로 길이를 나타냅니다.
 - `w`와 `h`는 1 이상 1,000 이하인 자연수입니다.

💡 알고리즘



완전탐색

😓 Code 성공

```
def solution(sizes):
    w = []
    h = []
    for i, j in sizes:
        if i >= j:
            w.append(i)
            h.append(j)
        continue
```

```
h.append(i)
w.append(j)
return max(w) * max(h)
```

고려사항

- 조금 더 효율적으로 짤 수 있는 방법 찾아보기! ⇒ 각 리스트 중 작은 것들 리스트에 담은 후 거기의 큰 값 * 각 리스트 중 큰 것들 리스트에 담은 후 큰 값

New!