

[문제 설명]

제로는 주식투자를 시작해서, 현재 종목을 여러개 가지고 있습니다.

최근 주식의 가격이 너무 떨어져서, 원금을 회수하고 투자를 그만두고 싶어서 종목을 정리하려고 합니다.

그래서 모든 종목의 가격을 0 으로 맞추고 그만두려고 하는데, 제로가 투자하는 주식시장은 조금 특이해서 아래와 같은 룰을 따릅니다.

1. 모든 종목을 1 주씩 구매해야 한다.
2. 선택한 종목보다 앞에 있는 모든 종목을 1 주씩 판매할 수 있다.
3. 선택한 종목보다 뒤에 있는 모든 종목을 1 주씩 판매할 수 있다.

예를 들어 만약 [10, 10, 4, 7] 네 종목을 제로가 보유하고 있고

제로가 4 를 선택했을 때 4 보다 앞에 있는 모든 수를 1 주씩 줄일 수 있고, 4 보다 뒤에 있는 모든 수를 1 주씩 줄일 수 있습니다.

또한, 보유종목 전체를 1 주씩 증가시킬 수 있습니다. 모든 종목을 0 주으로 만드는 최소 횟수를 반환해주세요.

위 예시에서는 첫 번째 10 을 선택하여 배열 전체를 4 주 감소시키고 나면, [6, 6, 0, 3]이 됩니다.

그 이후 두 번째 6 을 선택하여 두 번째 6 앞의 모든 종목을 6 주 감소시킵니다.

그러면 보유 종목 배열이 [0, 0, 0, 3]이 되고, 마지막으로 3 을 선택하여 3 뒤의 모든 종목을 3 주 감소시키면 [0, 0, 0, 0]이 됩니다.

그러면 $4 + 6 + 3$ 으로 13 을 반환해줍니다.

[제한 사항]

- 보유종목 배열의 길이는 1 이상 100,000 이하입니다.
- 각 종목들의 보유 주수는 -100,000 이상 100,000 이하입니다.

[입력 형식]

- 보유종목 배열인 stocks 가 주어집니다.

[출력 형식]

- 모든 종목을 0 으로 만들어 다 팔고 나올 수 있는 최소 작업 수를 반환해주세요.

제한 시간: 25 분

문제 유형: 그리디

난이도: 중

매개변수

stocks
Array / list

리턴타입

리턴타입
int

초기코드 Python

```
def solution(stocks):  
    ...  
    :param stocks: list  
    :return: int  
    ...  
    answer = 0  
  
    return answer
```

초기코드 JavaScript

```
/**  
 * @param {array} stocks  
 * @return {int}  
 */  
function solution(stocks) {  
    return 0  
}
```

테스트 케이스

예제용

입력값 grid
[4, -4, 4, -4]

출력값
36

채점용

입력값 grid	출력값
[-2, -2, -2]	2

