



그리디2

예시문제

ATM

잃어버린 괄호

예시문제

ATM

[문제]

인하은행에는 ATM이 1대밖에 없다. 지금 이 ATM앞에 N명의 사람들이 줄을 서있다. 사람은 1번부터 N번까지 번호가 매겨져 있으며, i번 사람이 돈을 인출하는데 걸리는 시간은 P_i 분이다.

사람들이 줄을 서는 순서에 따라서, 돈을 인출하는데 필요한 시간의 합이 달라지게 된다. 예를 들어, 총 5명이 있고, $P_1 = 3$, $P_2 = 1$, $P_3 = 4$, $P_4 = 3$, $P_5 = 2$ 인 경우를 생각해보자. [1, 2, 3, 4, 5] 순서로 줄을 선다면, 1번 사람은 3분만에 돈을 뽑을 수 있다. 2번 사람은 1번 사람이 돈을 뽑을 때 까지 기다려야 하기 때문에, $3+1 = 4$ 분이 걸리게 된다. 3번 사람은 1번, 2번 사람이 돈을 뽑을 때까지 기다려야 하기 때문에, 총 $3+1+4 = 8$ 분이 필요하게 된다. 4번 사람은 $3+1+4+3 = 11$ 분, 5번 사람은 $3+1+4+3+2 = 13$ 분이 걸리게 된다. 이 경우에 각 사람이 돈을 인출하는데 필요한 시간의 합은 $3+4+8+11+13 = 39$ 분이 된다.

줄을 [2, 5, 1, 4, 3] 순서로 줄을 서면, 2번 사람은 1분만에, 5번 사람은 $1+2 = 3$ 분, 1번 사람은 $1+2+3 = 6$ 분, 4번 사람은 $1+2+3+3 = 9$ 분, 3번 사람은 $1+2+3+3+4 = 13$ 분이 걸리게 된다. 각 사람이 돈을 인출하는데 필요한 시간의 합은 $1+3+6+9+13 = 32$ 분이다. 이 방법보다 더 필요한 시간의 합을 최소로 만들 수는 없다.

줄을 서 있는 사람의 수 N과 각 사람이 돈을 인출하는데 걸리는 시간 P_i 가 주어졌을 때, 각 사람이 돈을 인출하는데 필요한 시간의 합의 최솟값을 구하는 프로그램을 작성하시오.

[문제 해설]

문제에서 좀 이상하게 말했는데 그냥... P_i 가 적은 사람부터 정렬해서 누적으로 더하면 된다

[예시 코드]

```
N = int(input())
P = list(map(int, input().split()))
# P를 정렬한다
P.sort()
result = [0]
for i in range(N):
    # 제일 마지막으로 저장된 값 = 앞사람이 인출하는데까지 걸린 시간
    # 결과값으로 저장한 마지막 인덱스 + 현재 값을 쪽쪽 하면 된다
    result.append(result[-1] + P[i])
print(sum(result))
```

잃어버린 괄호

[문제]

세준이는 양수와 +, -, 그리고 괄호를 가지고 식을 만들었다. 그리고 나서 세준이는 괄호를 모두 지웠다.

그리고 나서 세준이는 괄호를 적절히 쳐서 이 식의 값을 최소로 만들려고 한다.

괄호를 적절히 쳐서 이 식의 값을 최소로 만드는 프로그램을 작성하시오.

[예제 입력]

55-50+40

[문제 해설]

값을 뺄 때 최대한 큰 값을 빼야 한다. 곱호 뒤부터 진행하면서 -를 만날 때까지 값을 더해나간다. -를 만나면 지금까지 더한 값을 음수로 바꿔서 계산을 한다

[예시 답안]

```
s = input()
arr = []
temp = ''
# 입력받은 값을 숫자, -, +로 나눈다
for i in s:
    if i.isnumeric():
        temp += i
    else:
```

```

    arr.append(temp)
    temp = ''
    arr.append(i)
arr.append(temp)

# 값을 계산하기 위해 스택에 숫자를 넣고 빼고 반복할 것이다
stack = []
# 맨 마지막 인덱스, 즉 뒤에서부터 진행
idx = len(arr) - 1
while idx >= 0:
    # 숫자라면 스택에 int형으로 넣기
    if arr[idx].isnumeric():
        stack.append(int(arr[idx]))
    # +라면 스택에 들어가있는 값, 내 이전 인덱스의 숫자를 더한다
    elif arr[idx] == '+':
        stack.append(stack.pop() + int(arr[idx - 1]))
        idx -= 1
    # -라면 스택에 들어있던 값을 음수로 바꿔서 다시 넣는다
    else:
        stack.append(-stack.pop())
        idx -= 1

print(sum(stack))

```