Problem 3711: Maximum Transactions Without Negative Balance

Problem Information

Difficulty: Medium
Acceptance Rate: 51.31%
Paid Only: Yes
Tags: Array, Greedy, Heap (Priority Queue)

Problem Description

You are given an integer array `transactions`, where `transactions[i]` represents the amount of the `ith` transaction:

* A positive value means money is **received**. * A negative value means money is **sent**.

The account starts with a balance of 0, and the balance **must never become negative**. Transactions must be considered in the given order, but you are allowed to skip some transactions.

Return an integer denoting the **maximum number of transactions** that can be performed without the balance ever going negative.

Example 1:

Input: transactions = [2,-5,3,-1,-2]

Output: 4

Explanation:

One optimal sequence is `[2, 3, -1, -2]`, balance: `0 -> 2 -> 5 -> 4 -> 2`.

Example 2:

Input: transactions = [-1,-2,-3]

Output: 0

Explanation:

All transactions are negative. Including any would make the balance negative.

Example 3:

Input: transactions = [3,-2,3,-2,1,-1]

Output: 6

Explanation:

All transactions can be taken in order, balance: `0 -> 3 -> 1 -> 4 -> 2 -> 3 -> 2`.

Constraints:

* `1 <= transactions.length <= 105` * `-109 <= transactions[i] <= 109`

Code Snippets

C++:

```
class Solution {
public:
int maxTransactions(vector<int>& transactions) {

}
};
```

Java:

```
class Solution {
public int maxTransactions(int[] transactions) {

}
```

```
}
```

Python3:

```
class Solution:
def maxTransactions(self, transactions: List[int]) -> int:
```