**[Wine Buying and Selling](https://practice.geeksforgeeks.org/problems/wine-buying-and-selling/1)**

Given an array, Arr[] of size N represents N house built along a straight line with equal distance between adjacent houses. Each house has a certain number of wine and they want to buy/sell those wines to other houses. Transporting one bottle of wine from one house to an adjacent house results in one unit of work. The task is to find the minimum number of work is required to fulfill all the demands of those N houses.

1. if arr[i] < 0, then ith house wants to sell arr[i] number of a wine bottle to other houses.
2. if arr[i] > 0, then ith house wants to buy arr[i] number of a wine bottle from other houses.

**Note:** One have to print the minimum number such that, all the house can buy/sell wine to each other.  
It is guaranteed that sum of all the elements of the array will be 0.

**Example 1:**

**Input:** N = 5,

Arr[] = {5, -4, 1, -3, 1}

**Output:** 9

**Explanation:**

1th house can sell 4 wine bottles to 0th house,

total work needed 4\*(1-0) = 4, new arr[] = 1,0,1,-3,1

now 3rd house can sell wine to 0th, 2th and 4th.

so total work needed = 1\*(3-0)+1\*(3-2)+1\*(4-3) = 5

So total work will be 4+5 = 9

**Example 2:**

**Input:** N = 6,

Arr[]={-1000, -1000, -1000, 1000, 1000, 1000}

**Output:** 9000

**Explanation:**

0th house sell 1000 wine bottles to 3rd house,

total work needed 1000\*(3-0) = 3000.

1st house sell 1000 wine bottles to 4th house,

total work needed 3000 + 1000\*(3-0) = 6000.

2nd house sell 1000 wine bottles to 5th house,

total work needed 6000 + 1000\*(3-0) = 9000.

So total work will be 9000 unit.

**Your Task:**  
You don't need to read input or print anything. Complete the function **wineSelling()** which takes the array Arr[] and its size N as input parameters and returns an integer as output.

**Expected Time Complexity:**O(N)  
**Expected Auxiliary Space:** O(N)

**Constraints:**  
1 ≤ N ≤ 10^5  
-10^6 ≤ Arr[i] ≤ 10^6