문제

# ATM 성공

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **시간 제한** | **메모리 제한** | **제출** | **정답** | **맞은 사람** | **정답 비율** |
| 1 초 | 256 MB | 7711 | 5344 | 4582 | 70.341% |

## 문제

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

사람들이 줄을 서는 순서에 따라서, 돈을 인출하는데 필요한 시간의 합이 달라지게 된다. 예를 들어, 총 5명이 있고, P1 = 3, P2 = 1, P3 = 4, P4 = 3, P5 = 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 = 4분이 걸리게 된다. 3번 사람은 1번, 2번 사람이 돈을 뽑을 때까지 기다려야 하기 때문에, 총 3+1+4 = 8분이 필요하게 된다. 4번 사람은 3+1+4+3 = 11분, 5번 사람은 3+1+4+3+2 = 13분이 걸리게 된다. 이 경우에 각 사람이 돈을 인출하는데 필요한 시간의 합은 3+4+8+11+13 = 39분이 된다.

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

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

## 입력

첫째 줄에 사람의 수 N(1 ≤ N ≤ 1,000)이 주어진다. 둘째 줄에는 각 사람이 돈을 인출하는데 걸리는 시간 Pi가 주어진다. (1 ≤ Pi ≤ 1,000)

## 출력

첫째 줄에 각 사람이 돈을 인출하는데 필요한 시간의 합의 최소값을 출력한다.

## 예제 입력 1 복사

5  
3 1 4 3 2

## 예제 출력 1 복사

32

answer

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

import java.util.Arrays;

import java.util.StringTokenizer;

public class T\_11399 {

public static void main(String[] args) throws NumberFormatException, IOException {

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

int size = Integer.parseInt(br.readLine());

StringTokenizer st = new StringTokenizer(br.readLine());

int []times = new int[size];

for(int i=0;i<size;i++){

times[i] = Integer.parseInt(st.nextToken());

}

Arrays.sort(times);

int sum=0;

int total=0;

for(int i=0;i<size;i++){

sum += times[i];

total += sum;

}

System.out.println(total);

}

}