☰   🌸 Hello, **2024101067**.

File Compression

Submit solution

My submissions
All submissions
Best submissions

✔ **Points:** 100 (partial)
🕐 **Time limit:** 1.0s
🗏 **Memory limit:** 64M

⌄**Allowed languages**
   C

Modern file compression systems, like ZIP and TAR, often need to merge multiple files into a single archive efficiently. However, merging files takes time and computational resources, and the cost of merging two files is equal to the sum of their sizes.

To optimize performance, compression algorithms aim to minimize the total merging cost when combining all files into a single archive.

Task

You are designing an efficient file compression algorithm. Given an array arr[] where each element represents the size of a file (in MB), determine the minimum total cost required to merge all files into a single archive.

Your goal is to ensure that merging is as fast and efficient as possible, avoiding unnecessary computational overhead.

Input:

The first line of the input is n, the total number of files to be compressed. The second line contains n space separated integers a_i, the sizes of the files in MB.
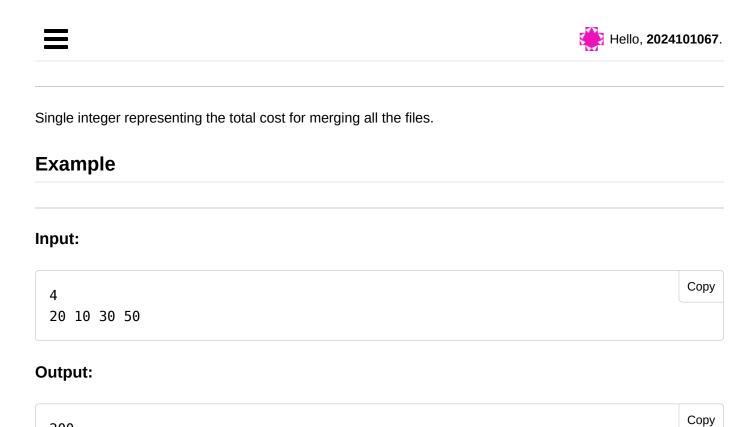
Constraints:

$$1 \le n \le 1e6$$
$$1 \le a_i \le 1e9$$

proudly powered by **DMOJ** | English (en) ⌄

Single integer representing the total cost for merging all the files.

Example

Input:

```
4
20 10 30 50
```

Copy

Output:

```
200
```

Copy

Explanation

Merge 10 MB and 20 MB → Cost = 10 + 20 = 30

Merge 30 MB and 30 MB → Cost = 30 + 30 = 60

Merge 50 MB and 60 MB → Cost = 50 + 60 = 110

Total Cost = 200

❓ Clarifications

Report an issue

No clarifications have been made at this time.