



# Add All

**Link submit:**

[https://uva.onlinejudge.org/index.php?option=onlinejudge&page=show\\_problem&problem=1895](https://uva.onlinejudge.org/index.php?option=onlinejudge&page=show_problem&problem=1895)

**Solution:**

C++	<a href="http://ideone.com/C0Qhit">http://ideone.com/C0Qhit</a>
Java	<a href="https://ideone.com/RkDjEG">https://ideone.com/RkDjEG</a>
Python	<a href="https://ideone.com/Es7EU0">https://ideone.com/Es7EU0</a>

**Tóm tắt đề:**

Cho bạn N số nguyên, bạn hãy tính chi phí để cộng toàn bộ các số nguyên này lại sao cho chi phí nhỏ nhất, các số nguyên được cộng mỗi lần cộng hai số riêng lẻ với nhau và lưu lại chi phí, hoặc có thể cộng một số vào một nhóm số đã được cộng trước đó, hoặc cũng có thể hai nhóm cộng lại với nhau:

Ví dụ bạn có 5 số N như sau: 2 2 2 2 3.

$2 + 2 = 4$  (chi phí 4)

$2 + 2 = 4$  (chi phí 4)

$3 + 4 = 7$  (chi phí 7, do số 3 nối với 2 + 2 ở đầu)

$4 + 7 = 11$  (chi phí 11, do nối 2 + 2 với 2 + 2 + 3)

Tổng chi phí để cộng tất cả các số trên là:  $4 + 4 + 7 + 11 = 26$ .

**Input:**

Có nhiều bộ test, mỗi bộ test có định dạng như sau:

- Dòng đầu tiên chứa số N ( $2 \leq N \leq 5000$ ).
- Dòng tiếp theo lần lượt chứa N số nguyên không vượt quá 100000.

Kết thúc các bộ test là số 0 ở cuối cùng.

**Output:**

Mỗi bộ test in ra số duy nhất là chi phí nhỏ nhất để cộng tất cả N số lại với nhau.

**Ví dụ:**

5 2 2 2 2 3 3 1 2 3	26 9
------------------------------	---------

**Hướng dẫn giải:**

Bạn sẽ bỏ toàn bộ N số vào hàng đợi ưu tiên, mỗi lần cộng hai số hoặc hai nhóm nào lại với nhau bạn sẽ cộng dồn chi phí lại, đồng thời lấy hai số/nhóm đó ra khỏi hàng đợi. Chi phí mới nhận được lại tiếp tục bỏ vào trong hàng đợi ưu tiên.

Khi hàng đợi ưu tiên không còn số nào thì kết quả cộng các chi phí lại sẽ là kết quả cần tìm.

**Độ phức tạp:**  $O(N\log N)$  với N là chi phí xét toàn bộ các cặp số trong N số,  $\log N$  là chi phí push/pop phần tử vào trong hàng đợi ưu tiên.