



# Find the Median

Link submit: <https://www.hackerrank.com/challenges/find-the-median>

Solution:

C++	<a href="http://ideone.com/FhYky7">http://ideone.com/FhYky7</a>
Java	<a href="https://ideone.com/XcSUUs">https://ideone.com/XcSUUs</a>
Python	<a href="https://ideone.com/kG6lku">https://ideone.com/kG6lku</a>

Tóm tắt đề:

Cho một dãy số, tìm phần tử trung vị của dãy số đó.

Input:

Dòng đầu tiên chứa số nguyên  $n$  ( $1 \leq n \leq 1000001$ ,  $n$  lẻ) là số lượng phần tử của dãy số.

Dòng thứ hai chứa  $n$  số nguyên là các phần tử của dãy có giá trị thuộc đoạn  $[-10000, 10000]$ .

Output:

Xuất ra phần tử trung vị của dãy đó.

Ví dụ:

7	3
0 1 2 4 6 5 3	

Giải thích ví dụ:

Nếu ta viết dãy số theo thứ tự tăng dần thì sẽ thu được dãy  $\{0, 1, 2, 3, 4, 5, 6\}$ , phần tử nằm ở vị trí chính giữa dãy là phần tử 3.

Hướng dẫn giải:

Ý tưởng bài này khá đơn giản, vì phần tử trung vị là phần tử nằm giữa mảng khi sắp xếp lại, đặt biệt trong bài này mảng luôn có số lượng phần tử là lẻ nên sẽ chỉ có 1 phần tử trung vị. Vậy thì ta sẽ sắp xếp lại mảng tăng dần (hoặc giảm dần). Sau đó lấy phần tử ở vị trí  $n/2$  ra là được (vị trí tính từ 0, nếu ta lưu vào mảng có vị trí tính từ 1 thì chỉ cần lấy phần tử  $n/2+1$ ).

**Độ phức tạp:**  $O(n \log n)$  với  $n$  là số lượng phần tử, trong đó chi phí sắp xếp tốn  $O(n \log n)$ , việc lấy phần tử ở vị trí  $n/2$  chỉ  $O(1)$ .