



✓ [Mảng 1 Chiều Nâng Cao]. Bài 5. Gửi thư

[C](#)[C++ CB NC](#)[C++ NC](#)[DSA](#)[JAVA OOP](#)[Xem dạng PDF](#)

Tất cả các thành phố của **Lineland** đều nằm trên trục tọa độ Ox. Do đó, mỗi thành phố được liên kết với vị trí xi - tọa độ trên trục Ox. Không có hai thành phố được đặt tại một điểm. Cư dân **Lineland** thích gửi thư cho nhau. Một người chỉ có thể gửi thư nếu người nhận sống ở một thành phố khác. Chi phí gửi thư chính xác bằng khoảng cách giữa thành phố của người gửi và thành phố của người nhận. Đối với mỗi thành phố, hãy tính hai giá trị **mini** và **maxi**, trong đó **mini** là chi phí tối thiểu để gửi thư từ thành phố thứ i đến một thành phố khác và **maxi** là chi phí tối đa để gửi thư từ thành phố thứ i đến một số thành phố khác

Gợi ý : **mini** = khoảng cách nhỏ nhất giữa 2 thành phố đứng trước và đứng sau của thành phố hiện tại, **maxi** = khoảng cách lớn hơn tính từ thành phố hiện tại tới thành phố đầu tiên và từ thành phố hiện tại tới thành phố cuối cùng. Vì mảng đã sắp tăng dần nên mới có điều này.

Đầu vào

Dòng đầu tiên là số nguyên dương n Dòng thứ hai chứa chuỗi n số nguyên khác nhau x1, x2, ..., xn ($-10^9 \leq x_i \leq 10^9$), trong đó xi là tọa độ x của thành phố thứ i.

Tất cả các xi là khác biệt và theo thứ tự tăng dần.

Giới hạn

$$2 \leq n \leq 10^6$$

$$-10^9 \leq x_i \leq 10^9$$

Đầu ra

Đối với mỗi thành phố in ra 2 giá trị **mini** và **maxi** trên 1 dòng.

[Gửi bài giải](#)

[Bài nộp của tôi](#)
[Danh sách bài nộp](#)
[Bài nộp tốt nhất](#)

✓ **Điểm:** 1,00 (OI)

🕒 **Giới hạn thời gian:** 1.0s

📄 **Giới hạn bộ nhớ:** 256M

🖨 **Input:** stdin

🖨 **Output:** stdout

📝 **Nguồn bài:**
28Tech

➤ **Dạng bài**

✓ **Ngôn ngữ cho phép**
C, C#, C++, Java, Kotlin,
Pascal, PyPy, Python,
Scratch



Input 01

```
4
-5 -2 2 7
```

[Copy](#)

Output 01

```
3 12
3 9
4 7
5 12
```

[Copy](#)

Bình luận

[Báo cáo vấn đề](#)

Hãy đọc [nội quy](#) trước khi bình luận.

Không có bình luận tại thời điểm này.