

Bài 1. Trộn 2 dãy

◆ Bài toán:

Cho hai mảng A và B có độ dài lần lượt là N và M, đều đã được **sắp xếp tăng dần**. Nhiệm vụ của bạn là trộn hai mảng này lại thành một mảng mới và **vẫn giữ nguyên thứ tự tăng dần**.

Yêu cầu thuật toán có **độ phức tạp tuyến tính ($O(N + M)$)**.

◆ Input:

- Dòng đầu tiên: hai số nguyên `N` và `M` — số lượng phần tử của mảng A và B.
- Dòng thứ hai: `N` số nguyên tăng dần — các phần tử của mảng A.
- Dòng thứ ba: `M` số nguyên tăng dần — các phần tử của mảng B.

◆ Output:

- In ra mảng mới sau khi trộn hai mảng, theo thứ tự tăng dần.

◆ Ràng buộc:

- `1 ≤ N, M ≤ 107`
- `1 ≤ A[i], B[i] ≤ 109`

◆ Ý tưởng giải:

Sử dụng kỹ thuật **two pointer (hai con trỏ)**:

- Duyệt song song hai mảng A và B bằng hai biến chỉ số `i` và `j`.
- So sánh `A[i]` và `B[j]`, phần tử nhỏ hơn sẽ được đưa vào mảng kết quả.
- Tăng chỉ số tương ứng (i hoặc j).
- Sau khi duyệt hết một mảng, chép phần còn lại của mảng kia vào kết quả.

◆ Độ phức tạp:

- **Thời gian:** $O(N + M)$
- **Bộ nhớ:** $O(N + M)$

◆ Sample Input 0:

```
6 6
1 2 4 4 9 10
4 4 5 6 6 8
```

◆ Sample Output 0:

```
1 2 4 4 4 4 5 6 6 8 9 10
```