

Bài yêu cầu sắp xếp mảng ban đầu bằng thuật toán counting_sort

Thuật toán sắp xếp phân phối (Distribution Sort) là một nhóm các thuật toán sắp xếp dựa trên việc phân loại dữ liệu. Một ví dụ điển hình của thuật toán này là Radix Sort và Counting Sort. Dưới đây là mã C++ cho thuật toán Counting Sort:

```
void counting_sort(int A[], int n) {
    int max_val = *max_element(A, A + n); // lấy phần tử lớn nhất trong mảng A để tạo mảng tần suất phù hợp
    vector<int> count(max_val + 1, 0);

    for (int i = 0; i < n; i++)
        count[A[i]]++;

    int index = 0;
    for (int i = 0; i <= max_val; i++) {
        while (count[i] > 0) {
            A[index++] = i;
            count[i]--;
        }
    }

    for (int i = 0; i < n; i++)
        cout << A[i] << " ";
}
```

Lưu ý rằng Counting Sort chỉ hoạt động với các số nguyên không âm và khi giá trị lớn nhất trong mảng không quá lớn, do giới hạn về bộ nhớ. Nếu bạn muốn sắp xếp các số nguyên âm hoặc các số rất lớn, bạn có thể cần sử dụng một thuật toán sắp xếp khác như Quick Sort, Merge Sort hoặc Heap Sort.