

Bài yêu cầu sắp xếp 1 mảng sử dụng thuật toán Merge_Sort

Dưới đây là các bước của thuật toán sắp xếp trộn (Merge Sort):

1. Chia mảng thành hai nửa. Nếu mảng có nhiều hơn một phần tử, chúng ta chia mảng thành hai nửa.
2. Sắp xếp mỗi nửa. Chúng ta sử dụng đệ quy để sắp xếp cả hai nửa của mảng.
3. Trộn hai nửa đã sắp xếp. Sau khi cả hai nửa đã được sắp xếp, chúng ta trộn chúng lại với nhau để có được mảng hoàn chỉnh đã được sắp xếp.

Ví dụ, với mảng 5 4 3 2 1, các bước sẽ như sau:

- Bước 1: Chia mảng thành hai nửa là 5 4 và 3 2 1.
- Bước 2: Sắp xếp cả hai nửa, ta được 4 5 và 1 2 3.
- Bước 3: Trộn hai nửa đã sắp xếp, ta được 1 2 3 4 5.

Và cuối cùng, sau tất cả các bước, mảng đã được sắp xếp theo thứ tự tăng dần.

Dưới đây là mã C++ cho thuật toán Merge Sort:

```
// Hàm trộn hai mảng con đã sắp xếp
void merge(int A[], int left, int mid, int right) {
    int i, j, k;
    int n1 = mid - left + 1;
    int n2 = right - mid;

    // Tạo mảng tạm
    vector<int> L(n1), R(n2);

    // Copy dữ liệu vào mảng tạm
    for (i = 0; i < n1; i++)
        L[i] = A[left + i];
    for (j = 0; j < n2; j++)
        R[j] = A[mid + 1 + j];

    // Trộn hai mảng tạm vào mảng A
    i = 0;
    j = 0;
    k = left;
    while (i < n1 && j < n2) {
        if (L[i] <= R[j]) {
            A[k] = L[i];
            i++;
        } else {
            A[k] = R[j];
            j++;
        }
        k++;
    }
    while (i < n1)
        A[k++] = L[i++];
    while (j < n2)
        A[k++] = R[j++];
}
```

```

        A[k] = R[j];
        j++;
    }
    k++;
}

// Copy các phần tử còn lại của L[], nếu có
while (i < n1) {
    A[k] = L[i];
    i++;
    k++;
}

// Copy các phần tử còn lại của R[], nếu có
while (j < n2) {
    A[k] = R[j];
    j++;
    k++;
}
}

// Hàm thực hiện thuật toán Merge Sort
void merge_sort(int A[], int left, int right) {
    if (left < right) {
        // Tìm điểm chính giữa của mảng
        int mid = left + (right - left) / 2;

        // Đệ quy tiếp tục chia đôi và sắp xếp
        merge_sort(A, left, mid);
        merge_sort(A, mid + 1, right);

        // Trộn hai nửa đã sắp xếp
        merge(A, left, mid, right);
    }
}

```