

Thuật toán Duyệt là gì?

Thuật toán Duyệt (Brute force) là một phương pháp giải quyết vấn đề bằng cách thử hết tất cả các trường hợp có thể xảy ra và tìm ra kết quả tối ưu dựa trên kết quả của mỗi trường hợp. Tuy nhiên, với các bài toán có quy mô lớn, việc thử hết tất cả các trường hợp sẽ rất tốn thời gian và không khả thi.

Phương pháp Duyệt thường được sử dụng để kiểm tra tính đúng đắn của một giải pháp hoặc để tìm kiếm giải pháp tối ưu cho các bài toán quy mô nhỏ.

Ví dụ về việc áp dụng thuật toán Duyệt là bài toán tìm tất cả các tập con của một tập hợp. Giả sử bạn có tập hợp {1, 2, 3} và bạn muốn tìm tất cả các tập con của tập hợp này. Tập hợp tất cả các tập con của tập hợp {1, 2, 3} là: {}, {1}, {2}, {3}, {1, 2}, {1, 3}, {2, 3}, {1, 2, 3}.

Dưới đây là đoạn mã C++ áp dụng phương pháp Duyệt để tìm tất cả các tập con của một tập hợp:

```
#include <iostream>
#include <vector>

using namespace std;

void liet_ke_tap_con(vector<int>& tap_ho, vector<int>& tap_con, int
vi_tri) {
    if (vi_tri == tap_ho.size()) {
        cout << "{";
        for (int i = 0; i < tap_con.size(); i++) {
            if (i > 0) {
                cout << ", ";
            }
            cout << tap_con[i];
        }
        cout << "}" << endl;
        return;
    }

    tap_con.push_back(tap_ho[vi_tri]);
    liet_ke_tap_con(tap_ho, tap_con, vi_tri + 1);
    tap_con.pop_back();
    liet_ke_tap_con(tap_ho, tap_con, vi_tri + 1);
}

int main() {
    vector<int> tap_ho = {1, 2, 3};
    vector<int> tap_con;
    liet_ke_tap_con(tap_ho, tap_con, 0);
    return 0;
}
```

Cho thêm 1 ví dụ khác và viết mã bằng C++

Ví dụ khác về thuật toán Duyệt là bài toán tìm chuỗi con có tổng lớn nhất của một mảng số nguyên. Giả sử bạn có một mảng số nguyên {1, -2, 3, 4, -1, 2, -6, 5} và bạn muốn tìm chuỗi con có tổng lớn nhất của mảng này. Chuỗi con có tổng lớn nhất là {3, 4, -1, 2, -6, 5}, với tổng là 7.

Dưới đây là đoạn mã C++ áp dụng phương pháp Duyệt để tìm chuỗi con có tổng lớn nhất của một mảng số nguyên:

```
#include <iostream>
#include <vector>

using namespace std;

int tim_chuoi_con_tong_lon_nhat(vector<int> a) {
    int max_tong = a[0];
    int tong = 0;
    int n = a.size();
    for (int i = 0; i < n; i++) {
        for (int j = i; j < n; j++) {
            tong += a[j];
            cout << a[j] << " ";
        }
        if (tong > max_tong) {
            max_tong = tong;
        }
        cout << " | Tong: " << tong << ", max Tong: " << max_tong <<
endl;
        tong = 0;
    }
    return max_tong;
}

int main() {
    vector<int> a = {1, -2, 3, 4, -1, 2, -6, 5};
    int tong_lon_nhat = tim_chuoi_con_tong_lon_nhat(a);
    cout << "Chuoi con co tong lon nhat la: " << tong_lon_nhat <<
endl;
    return 0;
}
```