

Tìm kiếm độ sâu lặp đi lặp lại (IDDFS)

Tìm kiếm độ sâu lặp đi lặp lại

Giải thích thuật toán:

Tìm kiếm độ sâu lặp đi lặp lại (IDDFS) kết hợp các lợi ích của tìm kiếm ưu tiên độ sâu và tìm kiếm ưu tiên chiều rộng.

Nó sử dụng tìm kiếm giới hạn độ sâu (DLS) như một quy trình con và lặp đi lặp lại giới hạn độ sâu cho đến khi tìm thấy giải pháp.

Thuật toán đảm bảo tính đầy đủ, vì nó kiểm tra các nút ở mỗi cấp độ. Nó kết thúc khi một giải pháp được phát hiện hoặc khi tất cả các cấp độ được khám phá mà không tìm thấy giải pháp.

Các bước chính:

1. Bắt đầu với độ sâu = 0.
2. Thực hiện Tìm kiếm giới hạn độ sâu đến độ sâu hiện tại.
3. Nếu tìm thấy mục tiêu, trả lại giải pháp; nếu không, hãy tăng độ sâu.

Tìm kiếm giới hạn độ sâu khám phá các đường dẫn trong giới hạn độ sâu hiện tại, trả về 'giới hạn' cho các nút vượt quá giới hạn đó.

Mã giả:

function ITERATIVE-DEEPENING-SEARCH(problem) trả về một giải pháp hoặc thất bại

Tìm kiếm độ sâu lặp đi lặp lại (IDDFS)

đối với độ sâu = 0 đến vô cực do kết quả <- DEPTH-

LIMITED-SEARCH(vấn đề, độ sâu)

nếu result != cutoff thì trả về kết quả

Triển khai Python:

Việc triển khai Python sử dụng lớp Node để biểu diễn các trạng thái. Một hàm đệ quy thực hiện tìm kiếm giới hạn độ sâu (DLS), trong khi hàm lặp lại tăng dần giới hạn độ sâu. Mỗi bước theo dõi các nút đã truy cập, độ sâu và giới hạn.

Tham khảo mã nguồn để biết chi tiết.

Ví dụ đầu ra:

Độ sâu: 0, Dấu vết: [('Arad', 0)]

Độ sâu: 1, Dấu vết: [('Arad', 1), ('Zerind', 0), ('Sibiu', 0), ('Timisoara', 0)]

Độ sâu: 2, Dấu vết: [('Arad', 2), ('Zerind', 1), ('Oradea', 0), ('Sibiu', 1), ...]

Giải pháp được tìm thấy: ['Arad', 'Sibiu', 'Fagaras', 'Bucharest']