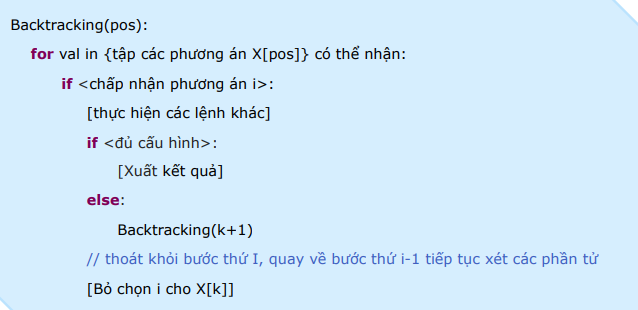
**Ôn tập các thuật toán đã học**

**Backtracking**

Backtracking được sử dụng khi giải quyết các vấn đề tập hợp mà ta phải kiểm tra tất cả các phương án giải quyết (vét cạn)

* Khi vấn đề có thể được hình thành dưới dạng tìm kiếm một giải pháp từ một tập hợp các ứng cử viên.
* Xây dựng từng bước giải quyết vấn đề



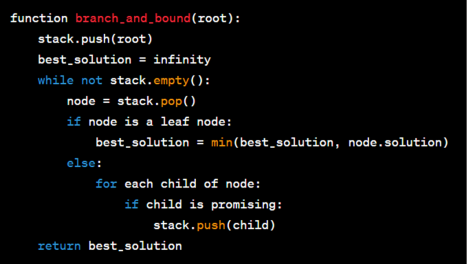
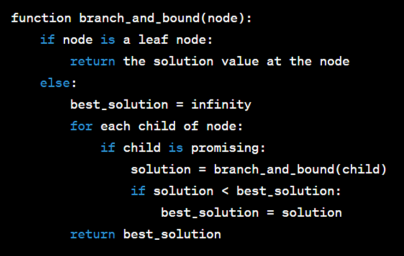
**Branch and bound**

Branch and bound được sử dụng để giải quyết các vấn đề tối ưu hóa, tìm ra câu trả lời cho vấn đề từ một tập các phương án tiềm năng

* Vấn đề có thể được biểu diễn dưới dạng cây tìm kiếm, trong đó mỗi nút đại diện cho một giải pháp từng phần.
* Bài toán có hàm mục tiêu để kiểm tra phương án từng phần đó còn hợp lệ không
* Bài toán có ràng buộc để bỏ qua sớm các phương án không hợp lệ

DFS (đệ quy)

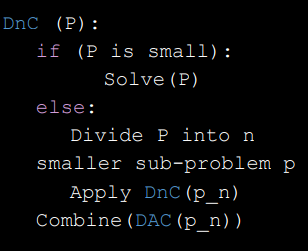
DFS(stack)



**Divide and conquer**

Chia để trị là một thuật toán giải quyết các vấn đề có thể chia được thành các vấn đề nhỏ hơn giống nhau

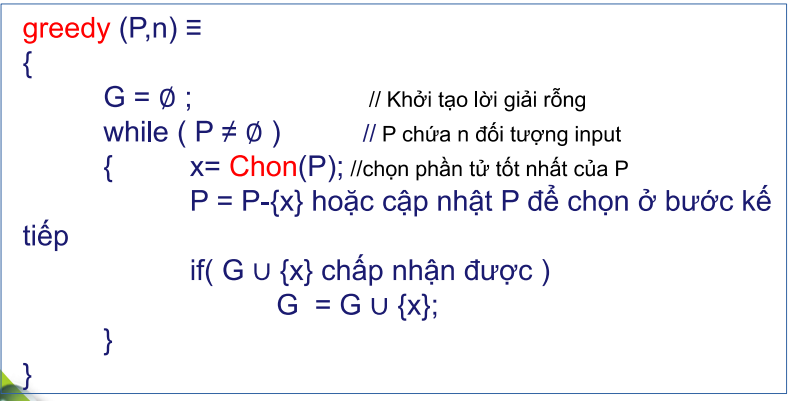
* Vấn đề có thể được chia thành các vấn đề con độc lập với nhau
* Các vấn đề con có kích thước nhỏ hơn và dễ giải quyết hơn
* Kết hợp các phương án giải quyết vấn đề con sẽ tạo thành phương án cho bài toán gốc



**Greedy approach**

Bài toán tham lam được sử dụng để giải quyết bài toán tối ưu hóa nơi ta chọn phương án tối ưu nhất ở từng bước được sử dụng khi

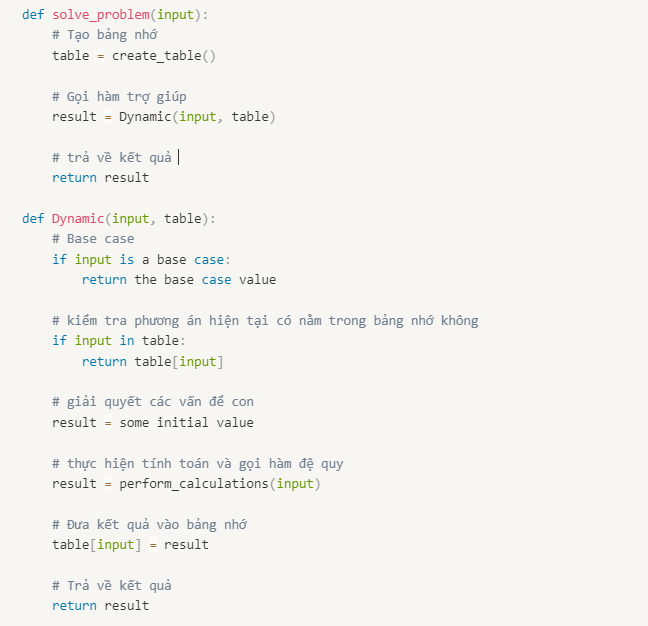
* Vấn đề có chứa thuộc tính tối ưu từng bước, việc chọn phương án tối ưu ở từng bước sẽ dẫn tối phương án tối ưu cho cả bài toán
* Bài toán có vấn đề con giải quyết độc lập với nhau và phương án tối ưu toàn được được xây dựng từng bước



**Dynamic programming**

Quy hoạch động được sử dụng để giải quyết bài toán tối ưu có chứa các vấn đề con chồng chất lên nhau và có cấu trúc tối ưu được sử dụng khi

* Vấn đề có thể được chia thành các vấn đề con chồng chất lên nhau ( để giải quyết vấn đề hiện tại cần giải quyết vấn đề nhỏ tương ứng liên quan tới nó vd: Dp[2] = Dp[1] + Dp[0] )
* Phương án toàn cục được xây dựng từ các vấn đề con



**Geometric algorithms**

Bài toán được sử dụng để giải quyết các vấn đề liên quan đến cấu trúc hình học.

* Giải quyết các bài toán liên quan đến hình học như điểm, đường thẳng, đa giác, …
* Các vấn đề thường được giải quyết là đường thẳng cắt nhau, biến đổi hình học, convex hull, ..
* Có thể sử dụng các thuật toán tối ưu đã học để giải quyết

**Graph algorithms**

Bài toán được sử dụng để giải quyết các bài toán đồ thị.

* Vấn đề được biểu diễn ở dạng đồ thị, các nút đại diện cho vật thể, các cạnh biểu diễn cho quan hệ giữa các vật thể
* Vấn đề yêu cầu phải duyệt hoặc tìm kiếm biểu đồ để tìm đường đi, chu trình hoặc cấu trúc cụ thể bên trong nó.
* Yêu cầu thực hiện các tác vụ tối ưu đồ thị như spanning tree, đường đi ngắn nhất,..