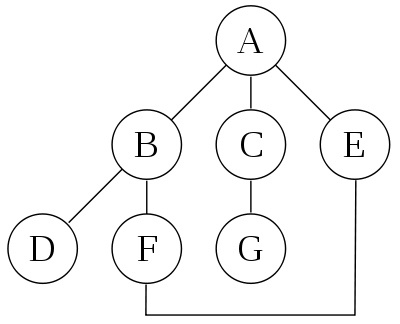
# Câu 1: Trình bày nguyên lý của giải thuật tìm kiếm theo chiều sâu, sơ đồ duyệt, cho ví dụ

*Nguyên lý:*  là một dạng tìm kiếm không đầy đủ mà quá trình tìm kiếm từ đỉnh con đầu tiên cho tới khi gặp được đỉnh cần tìm hoặc tới một nút không có con. Khi đó giải thuật quay lui về đỉnh vừa mới tìm kiếm ở bước trước.

*Ví dụ sơ đồ duyệt:* Quá trình tìm kiếm các đỉnh diễn ra như sau: bắt đầu đỉnh A, vì B chưa duyệt nên theo cạnh AB ta đến điểm B, tiếp tục theo cạnh BD tới D. Từ D không thể tiếp tục đi xa hơn, ta quay lại B. Từ B, theo BF đến điểm F, từ F đến điểm E. vì A đã đến điểm cuối cùng là điểm E nên ta quay điểm F, rồi quay lại B. Từ B đã xem xét nên ta quay lại A. Quá trình từ điểm A tiếp tục với các đỉnh C và G.

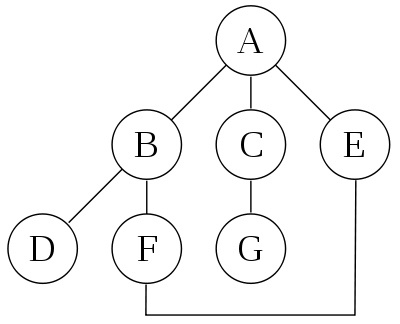
Độ phức tạp: O(|V| + |E|), V đỉnh duyệt qua, E cạnh duyệt qua

# Câu 2: Trình bài nguyên lý tìm kiếm sâu dần , cho ví dụ sơ đồ duyệt

*Nguyên lý*: thuật toán tìm kiếm kiếm sâu dần là một biến thể khác của tìm kiếm theo chiều sâu và tìm kiếm giới hạn theo chiều sâu

Gồm 2 bước cơ bản:

1. Tìm kiếm giới hạn chiều sâu.

2. Nếu tìm không tìm được kết quả thì tăng chiều sâu và tìm lại

*Ví dụ sơ đồ duyệt:* bắt đầu từ cấp 0 tới cấp 1 từ A qua trái gặp B tới C tới E kết thúc tìm kiếm trở về . từ A tới B . từ B tới D từ D tới F do hết duyệt từ F tới C từ C tới G từ G đến E khi hết trả về F . đến lượt đi theo A tới B và B tới D từ D tới F từ F đến E ta trở về cấp 1 từ R đến C từ C đến G từ G đến E từ E đến F tới B

ĐPT:

# Câu 3: Trình bài 4 bước nguyên tắc chuẩn tắc trong logic mệnh đề

# Câu 4: Trình bài 8 bước nguyên tắc chuyển dạng chuẩn tắc trong logic vị từ

* QT1: Loại bỏ ⇔: thay thế α ⇔ β bằng (α ⇒ β)∧(β ⇒ α).
* QT2: Loại bỏ ⇒: Thay thế α ⇒ β bằng ¬α ∨ β
* QT3: chuyển hoặc loại bỏ dấu ¬ đặt trước các ký hiệu bằng các luật deMorgan và luật phủ định kép

¬(α∨β)= ¬α ∧ ¬β;

¬(α∧β)= ¬α ∨ ¬β;

¬¬α= α;

¬∀xP(x)= ∃x¬P(x);

¬∃xP(x)= ∀x¬P(x)

* QT4: Chuẩn hóa các biến: các biến lượng từ không được trùng tên,

ví dụ ∀xP(x) ∨ ∃xQ(x) chuyển thành ∀xP(x) ∨ ∃yQ(y)

* QT5: chuyển các lượng từ về đầu câu,

ví dụ ∀xP(x) ∨ ∃yQ(y) chuyển thành ∀x∃y P(x) ∨ Q(y)

* QT6: loại bỏ ∃ bằng giá trị vô danh:

ví dụ ∃x Rich(x) trở thành Rich(c) với c là ký hiệu hằng vô danh, không trùng với các ký hiệu có trong cơ sở tri thức. Chú ý khi ∃ đặt bên trong ∀, phải sử dụng hàm vô danh;

ví dụ: ∀x∃y∀z P(x,y,z) trở clause clause clause thành ∀x∀z P(x,f(x),z) với f là ký hiệu hàm vô danh, không trùng với ký hiệu hàm khác trong cơ sở tri thức.

* QT7: bỏ qua các ký hiệu lượng tử ∀
* QT8: Áp dụng luật phân phối của phép ∧ đối với phép

# Câu 5: Trình bài ký hiệu biểu diễn cú pháp của logic vị từ và logic mệnh đề

*Trình bài ký hiệu biểu diễn cú pháp của logic vị từ*

Logic vị từ cấp một sử dụng các loại ký hiệu sau đây.

Các ký hiệu hằng: a, b, c, An, Ba, John,...

Các ký hiệu biến: x, y, z, u, v, w,...

Các ký hiệu vị từ: P, Q, R, S, Like, Havecolor, Prime,...

Mỗi vị từ là vị từ của n biến (n ≥ 0). Chẳng hạn Like là vị từ của hai biến, Prime là vị từ một biến.

Các ký hiệu vị từ không biến là các ký hiệu mệnh đề.

Các ký hiệu hàm: f, g, cos, sin, mother, husband, distance,...

Mỗi hàm là hàm của n biến (n ≥ 1). Chẳng hạn, cos, sin là hàm một biến, distance là hàm của ba biến.

Các ký hiệu kết nối logic: ∧ (hội), ∨ (tuyển), ⎤ (phủ định), ⇒ (kéo theo), ⇔(kéo theo nhau).

Các ký hiệu lượng tử: ∀ (với mọi), ∃ (tồn tại).

Các ký hiệu ngăn cách: dấu phẩy, dấu mở ngoặc và dấu đóng ngoặc.

*Trình bài ký hiệu biểu diễn cú pháp của logic mệnh đề*

Các ký hiệu

* Hằng logic: True, False.
* Các ký hiệu mệnh đề (biến mệnh đề): P, Q, R, S...
* Các phép kết nối logic: ∧, ∨, ¬, ⇒, ⇔.
* Các dấu mở ngoặc ”(“ và đóng ngoặc ”)”.

• Các quy tắc xây dựng các công thức

Các biến mệnh đề là công thức. Nếu A và B là công thức thì:

(A^B) (đọc “A hội B” hoặc “A và B”)

(AvB) (đọc “A tuyển B”hoặc “A hoặc B”)

(⎤ A) (đọc “phủ định A”)

(A=>B) (đọc “A kéo theo B”hoặc “nếu A thì B”)

(A<=>B) (đọc “A và B kéo theo nhau”)

là các công thức.

# Câu 6: Trình bài chuẩn hội của logic vị từ

Các công thức tương đương có thể xem như các biểu diển khác nhau của cùng một sự kiện. Để

# Câu 7: Trình bài tìm kiếm sâu có giới hạn ưu điểm nhược điểm độ phức tạp

*Khái niệm:* Là một thuật toán phát triển các nút chưa xét các theo chiều sâu nhưng có giới hạn mức để tránh đi vào những con đường không mang lại kết quả tốt như trong thuật toán tìm kiếm sâu dần.

*Ưu điểm:*

* Tìm kiếm giới hạn độ sâu yêu cầu ít thời gian và không gian bộ nhớ hơn.
* Giải pháp sẽ được tìm thấy nếu nó tồn tại trong thời gian vô hạn.
* Chiến lược tìm kiếm thực hiện đệ quy xuống cây tìm kiếm.

*Nhược điểm:*

* Giới hạn độ sâu là bắt buộc để thuật toán này thực thi.
* Nút mục tiêu có thể không tồn tại trong giới hạn độ sâu đã đặt trước đó, điều này sẽ lặp lại thêm thời gian thực hiện.
* Nút mục tiêu sẽ không được tìm thấy nếu nó không tồn tại trong giới hạn mong muốn.

*Độ phước tạp:*

* Độ phức tạp thời gian được biểu thị như sau: Nó tương tự như DFS, tức là O (bl ), trong đó 1 là giới hạn độ sâu đã đặt.
* Độ phức tạp không gian được thể hiện như sau: Nó tương tự như DFS , là O (bl ), trong đó 1 là giới hạn độ sâu quy định.

# Câu 8: Trình bài nguyên lý tìm kiếm leo đồi ưu điểm nhược điểm độ phức tạp

*Khái niệm*: là một kỹ thuật tối ưu toán học thuộc họ tìm kiếm cục bộ. Nó thực hiện tìm một trạng thái tốt hơn trạng thái hiện tại để mở rộng. Để biết trạng thái tiếp theo nào là lớn hơn, nó dùng một hàm H để xác định trạng thái nào là tốt nhất.

*Ưu điểm*: Nhanh, đơn giản, hiệu quả trong không gian tìm kiếm nhỏ, ít lồi lõm.

*Nhược điểm*: Khó tìm thấy trạng thái đích nếu như không gian tìm kiếm lớn, có nhiều điểm tối ưu cục bộ.

*Độ Phước Tạp*: O(NlogN)