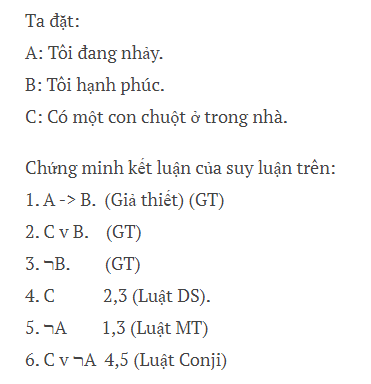
# NỘI DUNG ÔN TẬP THI TRÍ TUỆ NHÂN TẠO

## Hình thức hóa một bài toán suy diễn

**Ví dụ 1:** Nếu tôi đang nhảy thì tôi hạnh phúc. Có một con chuột ở trong nhà hoặc tôi hạnh phúc. Nhưng tôi buồn. Vì vậy, có một con chuột trong nhà và tôi không nhảy.



**Ví dụ 2:** Nếu An được lên chức và làm việc nhiều thì An sẽ được tăng lương. Nếu được tăng lương thì An sẽ mua xe mới. An không mua xe mới. Vì vậy, An không được lên chức hoặc không làm việc nhiều.

Ta đặt:

“An được lên chức” là p

“An làm việc nhiều” là q

“An sẽ được tăng lương” là r

“An sẽ mua xe mới” là t

“Nếu An được lên chức và làm việc nhiều thì An sẽ được tăng lương” ( p ^ q) → r

“Nếu được tăng lương thì An sẽ mua xe mới” r → t

“An không mua xe mới” ¬t

“An không được lên chức hoặc An không làm việc nhiều” ¬p v ¬q

Chứng minh: [( (p ^ q)→ r ) ^ ( r → t ) ^ ¬t ] => (¬p v ¬q)

( r → t )

¬t

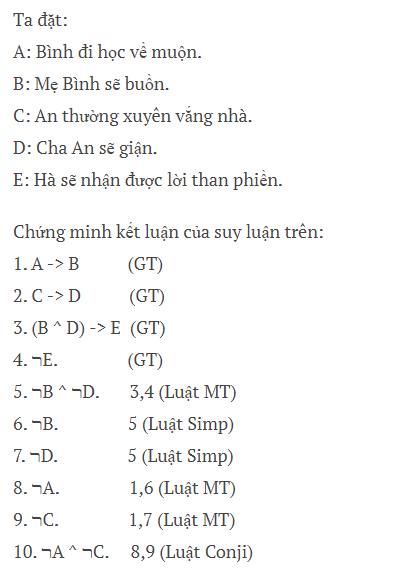
⸫ ¬r (Modus Tollens)

(p ^ q)→ r

⸫ ¬(p ^ q) (Modus Tollens)

¬p v ¬q (De Morgan)

**Ví dụ 3:** Nếu Bình đi học về muộn thì mẹ anh ta sẽ buồn. Nếu An thường xuyên vắng nhà thì cha anh ta sẽ giận. Nếu mẹ Bình buồn hoặc cha An giận thì cô Hà, bạn họ sẽ nhận được lời than phiền. Mà Hà không nhận được lời than phiền. Vì vậy, Bình đi học về sớm và An ít khi vắng nhà.



**Ví dụ 4:** Nếu muốn dự họp sáng thứ ba thì Minh phải dạy sớm. Nếu Minh đi nghe nhạc tối thứ hai thì Minh sẽ về trễ. Nếu về trễ và thức dậy sớm thì Minh phải đi họp mà chỉ ngủ dưới 7 giờ. Nhưng Minh không thể đi họp nếu chỉ ngủ dưới 7 giờ. Do đó hoặc là Minh không đi nghe nhạc thối thứ hai hoặc là Minh phải bỏ họp sáng thứ ba.

**Ví dụ 5:** Nếu Bình đi làm về muộn thì vợ anh ta sẽ rất giận dữ. Nếu An thường xuyên vắng nhà thì vợ anh ta sẽ rất giận dữ. Nếu vợ Bình hay vợ An giận dữ thì cô Hà bạn họ sẽ nhận được lời than phiền. Mà Hà không nhận được lời than phiền. Vậy Bình đi làm về sớm và An ít khi vắng nhà.

**Ta đặt:**

“Bình đi làm về muộn” là p

“Vợ Bình ta sẽ rất giận dữ” là q

“Nếu Bình đi làm về muộn thì vợ anh ta sẽ rất giận dữ” là p→q

“An thường xuyên vắng nhà” là r

“Vợ An ta sẽ rất giận dữ” là s

“Nếu An thường xuyên vắng nhà thì vợ anh ta sẽ rất giận dữ” là r → s

“Cô Hà bạn họ sẽ nhận được lời than phiền” là t

“Nếu vợ Bình hay vợ An giận dữ thì cô Hà bạn họ sẽ nhận được lời than phiền” là q ˅ s → t

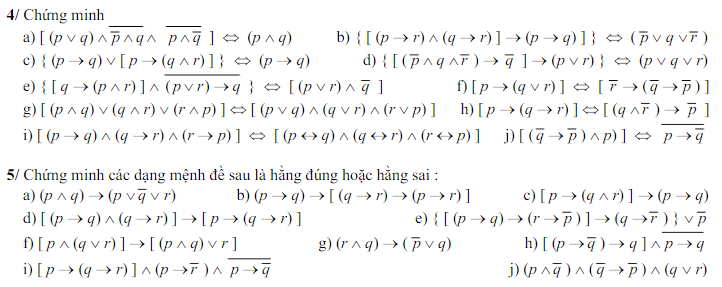
“Hà không nhận được lời than phiền” là ¬t

Chứng minh: [(p → q) ˄ (r → s) ˄ (q ˅ s → t) ˄ ¬t] ⇒ ¬p ˄ ¬r

1. p → q
2. r → s
3. q ˅ s → t
4. ¬t
5. ¬ (q ˅ s) (Modus Tollens của 3 và 4)
6. ¬q ˄ ¬s (De Morgan)
7. ¬p (Modus Tollens của 1 và 6)
8. ¬r (Modus Tollens của 2 và 6)
9. ¬p ˄ ¬r

## Các tương đương và suy diễn logic cơ bản

Có thể sử dụng bảng chân trị hoặc các quy tắc suy diễn để chứng minh các mệnh đề sau:



## Kiểm tra tính đúng của suy luận bằng thuật toán Vương Hạo









## Kiểm tra tính đúng của suy luận bằng thuật toán Robinson



Các bài ở trên giải bằng thuật toán này.

## Thuật toán suy diễn tiến

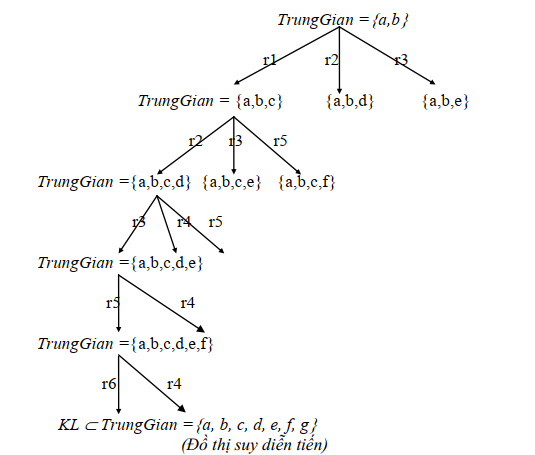
Cho trước tập các sự kiện giả thiết: GT = {a,b}.

Sử dụng tập RULE các luật:

r1: a → c, r2: b → d, r3: a → e, r4: a ∧ d → e, r5: b ∧ c → f,

r6: e ∧ f → g

Cần suy ra KL = {g}.



## Thuật toán suy diễn lùi

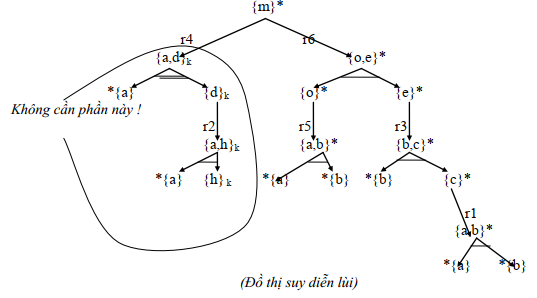
Cho trước tập các sự kiện giả thiết GT = {a, b}.

Sử dụng tập RULE các luật:

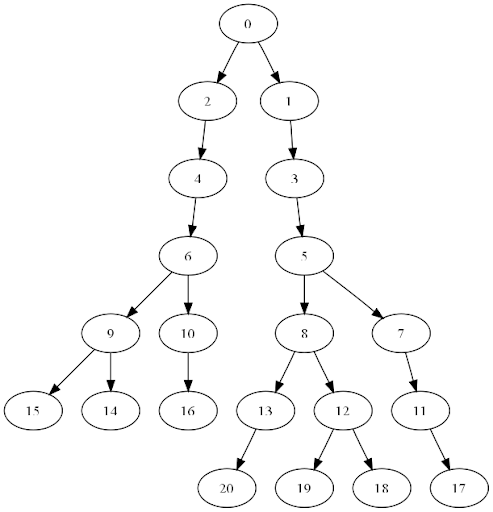
r1: a ∧ b → c, r2: a ∧ h → d, r3: b ∧ c → e, r4: a ∧ d → m

r5: a ∧ b → o, r6: o ∧ e → m

Cần suy ra KL = {m}.



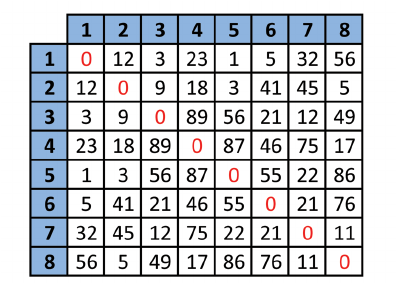
## Thuật toán tìm kiếm sâu dần với k = 3



## Chiến lược tham lam và bài toán người đi du lịch

Hãy tìm một hành trình cho một người giao hàng đi qua n điểm khác nhau, mỗi điểm đi qua một lần và trở về điểm xuất phát sao cho tổng chiều dài đoạn đường cần đi là ngắn nhất.

Các đỉnh đặt tên tương ứng là A, B, C, D, E, F, G, H



## Thuật giải AKT tìm kiếm cực tiểu với tri thức bổ sung trên cây, áp dụng vào bài toán đẩy số

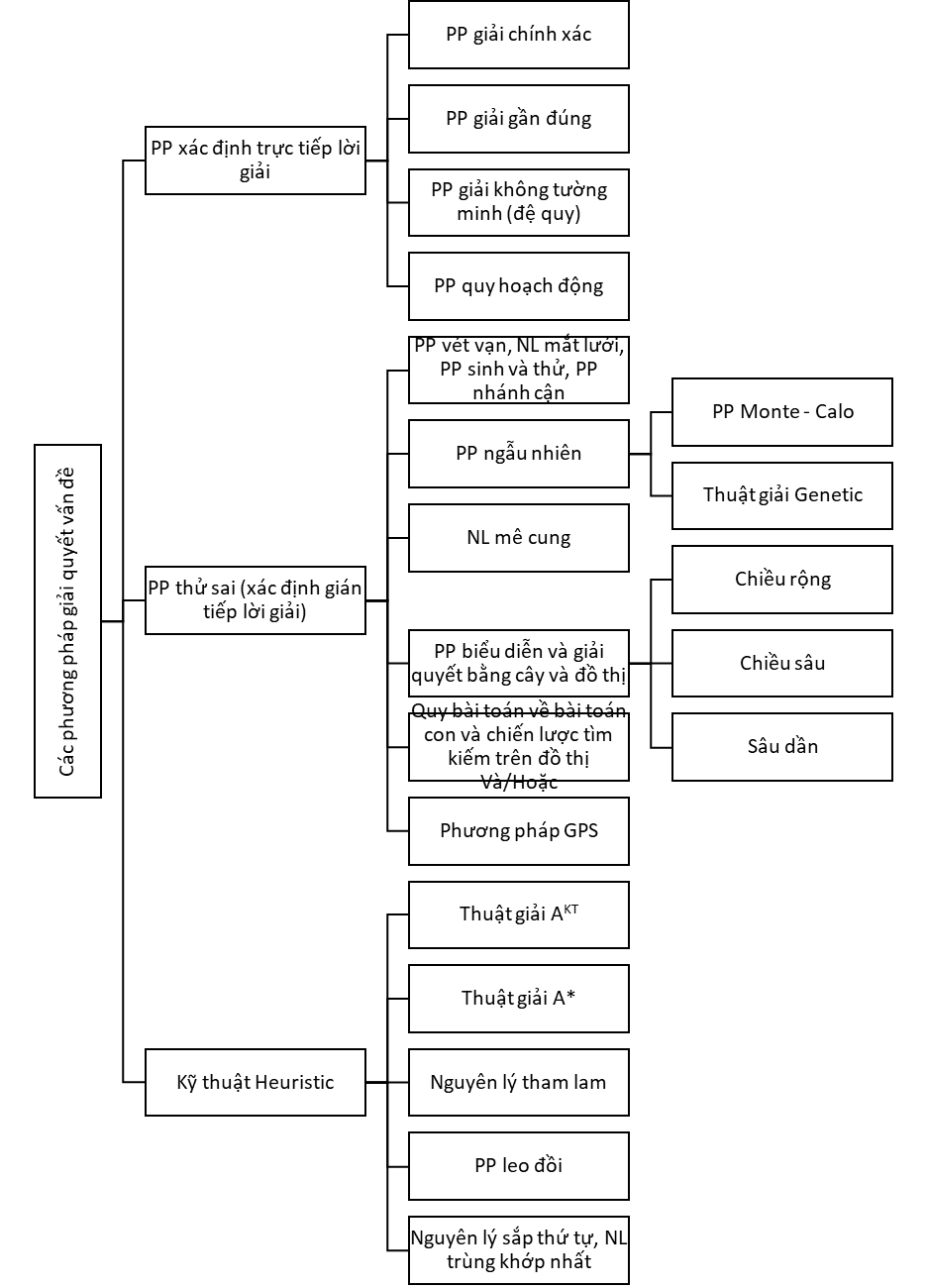
Xét bài toán đẩy số Tocci với n = 3 có hình trạng đầu x0 và hình trạng đích x\*, như sau:

A picture containing object, clock

Description automatically generated

Hàm heuristic h0(x) là số lượng chữ số (khác trống) trong trạng thái x không nằm ở đúng vị trí của chúng trong trạng thái đích x\*. Hãy minh họa từng bước áp dụng thuật giải AKT (tìm kiếm cực tiểu với tri thức heuristic bổ sung) để tìm dãy các bước đưa x0 và x\*.

## Tóm tắt các phương pháp tìm kiếm và giải quyết vấn đề



## Viết chương trình Prolog cơ bản sử dụng đệ quy, khử đệ quy và lát cắt: số fibonacci, giai thừa, tổ hợp, sinh số

Giai thừa:

domains

i = integer

predicates

factor(i, i)

clauses

factor(0, 1) :- !.

factor(N, KQ) :- N1=N-1, factor(N1, KQ1), KQ1\*N=KQ.

Tổ hợp:

domains

i = integer

predicates

combinatorial(i, i, i)

clauses

combinatorial(0, \_, 1) :- !.

combinatorial(K, K, 1) :- !.

combinatorial(K, N, KQ) :- N1=N-1, K1=K-1,

combinatorial(K, N1, KQ1),

combinatorial(K1, N1, KQ2),

KQ = KQ1+KQ2.

Fibonacci:

domains

i = integer

predicates

fibonacci(i, i)

clauses

fibonacci(1, 1) :-!.

fibonacci(2, 1) :-!.

fibonacci(N, KQ) :- N1 = N-1, N2= N-2,

fibonacci(N1, KQ1),

fibonacci(N2, KQ2),

KQ = KQ1 + KQ2.

Chia lấy nguyên:

domains

*int DivideWithoutRemainder(int divisor, int dividend)*

*{*

*int quotient = 0;*

*do*

*{*

*quotient++;*

*if (quotient \* dividend ≤ divisor &&*

*(quotient + 1) \* dividend > divisor)*

*return quotient;*

*} while (true);*

*}*

i = integer

predicates

Sinh\_1\_So(i)

ThuongNguyen(i, i, i)

clauses

Sinh\_1\_So(0).

Sinh\_1\_So(Sau) :- Sinh\_1\_So(Truoc), Sau = Truoc + 1.

ThuongNguyen(BC, C, KQ) :- Sinh\_1\_So(KQ),

KQ\*C <= BC,

(KQ+1) \* C > BC, !.

## Viết chương trình Prolog xử lý cơ bản trên danh sách: độ dài, tổng, nối, hoán vị, sắp xếp và tìm kiếm.

Tổng các phần tử trong danh sách:

domains

i = integer

ds = i\*

predicates

Tong(ds, i)

clauses

Tong([], 0) :- !.

Tong([H|T], KQ) :- Tong(T, KQ1),

KQ1 + H = KQ.

Độ dài danh sách:

domains

i = integer

ds = i\*

predicates

DoDai(ds, i)

clauses

DoDai([], 0).

DoDai([\_|T], KQ) :- DoDai(T, KQ1), KQ1 + 1 = KQ.

Nối 2 danh sách số nguyên:

domains

i = integer

ds = i\*

predicates

Noi(ds, ds, ds)

clauses

Noi([], L2, L2).

Noi([H1|T1], L2, [H1|T3]) :- Noi(T1, L2, T3).

Tìm các hoán vị của một danh sách số nguyên:

domains

i = integer

ds = i\*

predicates

Noi(ds, ds, ds)

HoanVi(ds, ds)

clauses

Noi([], L2, L2).

Noi([H1|T1], L2, [H1|T3]) :- Noi(T1, L2, T3).

HoanVi([], []).

HoanVi(L, [H|DuoiHV]) :-

Noi(Gi, [H|T], L), !,

Noi(Gi, T, ConLai),

HoanVi(ConLai, DuoiHV).

Sắp xếp danh sách: <http://kti.mff.cuni.cz/~bartak/prolog/sorting.html>

domains

i = integer

ds = i\*

predicates

i\_sort(ds, ds, ds)

insert\_sort(ds, ds)

insert(i, ds, ds)

clauses

insert\_sort(List, Sorted):-

i\_sort(List, [], Sorted).

i\_sort([], Acc, Acc).

i\_sort([H|T], Acc, Sorted):-

insert(H, Acc, NAcc),

i\_sort(T, NAcc, Sorted).

insert(X, [Y|T], [Y|NT]):-

X>Y,

insert(X, T, NT).

insert(X, [Y|T], [X,Y|T]):-

X<=Y.

insert(X, [], [X]).

Tìm kiếm phần tử trong danh sách:

domains

i = integer

ds = i\*

predicates

Search(i, ds)

clauses

Search(X,[X|\_]).

Search(X,[\_|T]):-

Search(X,T).

# ĐỀ THI LỚP VTK39

## Câu 1. Lý thuyết

### Vẽ sơ đồ cây phân loại các phương pháp giải quyết vấn đề đã học

### Có thể áp dụng những phương pháp giải quyết vấn đề nào để giải bài toán người giao hàng

“*Hàng ngày, người giao hàng phải chuyển hàng qua n địa điểm, mỗi địa điểm đúng một lần, rồi quay lại địa điểm xuất phát. Bài toán đặt ra là: làm thế nào để anh ta có được một hành trình với đường đi ngắn nhất*”

* **Phương pháp quy hoạch động**: Nghiệm của một bài toán là sự kết hợp các nghiệm của các bài toán con nhỏ hơn của nó (trong trường hợp rời rạc có tính đệ qui thì nghiệm trong n bước sẽ có được từ lời giải của k bước trước và lời giải trong n-k bước). Ta thường dùng phương pháp này để giải các bài toán tối ưu mà thỏa mãn nguyên lý trên.

Có thể xem nguyên lý tối ưu là một sự thể hiện tốt của phương pháp chia để trị trong việc giải quyết vấn đề. Khi thực hiện các tính toán trong phương pháp qui hoạch động, để thực hiện tính toán tại bước thứ n, nên *tận dụng các kết quả đã tính ở các buớc trước* thông qua các hệ thức truy hồi và một vài biến phụ để lưu các kết quả trung gian trước đó

* **Phương pháp tham lam**: Với thuật giải sử dụng nguyên lý tham lam giải bài toán tối ưu, ta sử dụng các thuật toán TìmKiếm với một ít điều chỉnh: chỉ chọn lời giải tốt nhất trong một bước kế tiếp (chứ chưa chắc tốt nhất trong cả quá trình), do đó không cài đặt cơ chế quay lui.

Ưu điểm của nguyên lý này là trong một số trường hợp, có thể nhanh thấy lời giải. Nhưng nhược điểm của nó là trong các trường hợp khác, có thể tồn tại lời giải tối ưu nhưng thuật toán này không tìm ra. Khi đó, do không dự phòng cơ chế quay lui nên có thể không tìm ra lời giải, mặc dù nó vốn tồn tại! (Tham cái lợi trước mắt mà không để ý đến cái lợi toàn cục và không dự phòng lưu lại đường đi đã qua để quay lui khi lạc hướng!)

## Câu 2. Áp dụng phương pháp tìm kiếm theo chiều rộng trên cây Và/Hoặc sau:

A picture containing object, clock

Description automatically generated

|  |  |
| --- | --- |
| Tập MO | Tập DONG |
| 1  2, 3  3, 4, 5  4, 5, 6, 7  5, 6, 7, 8, 9  6, 7, 8, 9, 12, 13  7, 8, 9, 12, 13, 10, 11 | 1  1, 2  1, 2, 3  1, 2, 3, 4  1, 2, 3, 4, 5  1, 2, 3, 4, 5, 6 (đỉnh 2, 3: “giải được”)  (Đỉnh gốc 1: “giải được” 🡪 Kết thúc |

## Câu 3. Xét bài toán đẩy số Tocci với n = 3 có hình trạng đầu x0 và hình trạng đích x\*, như sau:

A picture containing object, clock

Description automatically generated

Hàm heuristic h0(x) là số lượng chữ số (khác trống) trong trạng thái x không nằm ở đúng vị trí của chúng trong trạng thái đích x\*. Hãy minh họa từng bước áp dụng thuật giải AKT (tìm kiếm cực tiểu với tri thức heuristic bổ sung) để tìm dãy các bước đưa x0 và x\*.

**Giải**

Đầu tiên, ta gọi g0(x) = Độ dài đường đi hiện tại từ x0 đến x (số phép dịch chuyển ô trống đã đi để được trạng thái x)

h0(x) = số lượng chữ số (khác trống) trong trạng thái x không nằm ở đúng vị trí của chúng trong trạng thái đích x\*. Giá trị nằm ở góc trên bên trái của mỗi trạng thái.

Do đó, f0(x) = g0(x) + h0(x). Giá trị nằm ở góc trên bên phải của mỗi trạng thái.

## Câu 4. Kiểm tra tính hằng đúng của các biểu thức sau

1. bằng thuật toán Vương Hạo
2. bằng thuật toán Robinson
3. bằng một trong hai thuật toán trên

## Câu 5.

1. Áp dụng thuật toán suy diễn tiến kiểm tra rằng
2. Áp dụng thuật toán suy diễn lùi kiểm tra rằng từ GT = {a, b} có thể suy ra KL = {g} qua tập RULE = hay không?

# ĐỀ THI LỚP CTK39

## Câu 1.

### Vẽ sơ đồ cây phân loại các phương pháp giải quyết vấn đề đã học

### Có thể áp dụng những phương pháp giải quyết vấn đề nào để giải bài toán “Liệt kê tất cả các xâu nhị phân độ dài n”. Nêu ra ý tưởng áp dụng và cho một ví dụ minh họa

* Bắt đầu từ dãy các số 0
* ***Bước 1***. Chạy từ sau đến trước (từ phải sang trái), gặp số 0 đầu tiên thì dừng.
* ***Bước 2***. Đổi số 0 tại vị trí đó thành 1, các số khác ở trước nó giữ nguyên, các phần tử nằm sau nó thành 0 hết.
* ***Bước 3***. In kết quả ra màn hình,.
* ***Bước 4***. Nếu cả dãy số chưa phải là 1 hết thì quay lại bước 1, còn không thì dừng.

0 0 0 0 0

0 0 0 0 1

0 0 0 1 0

0 0 0 1 1

0 0 1 0 0

0 0 1 0 1

0 0 1 1 0

0 0 1 1 1

0 1 0 0 0

(Thật ra nó là dãy nhị phân tăng 1 đơn vị ở mỗi bước thôi)

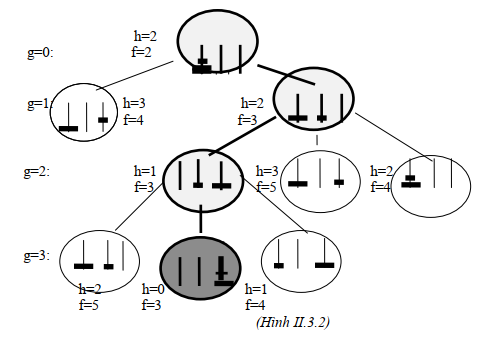
## Câu 2.

### Hãy tìm ra đường đi ngắn nhất (minh họa từng bước thực hiện qua các tập Đóng, Mở) bằng chiến lược tìm kiếm sâu dần với k = 2 từ “A” đến “I” hoặc “E” trên cây cho trong hình dưới đây, với giá thành được in nghiêng trên trong cung cung nối tương ứng, chẳng hạn giá đu từ A đến B là *15*.

A clock in the middle of a watch

Description automatically generated

### Áp dụng thuật toán tìm kiếm cực tiểu với tri thức bổ sung giải bài toán Tháp Hà Nội với số đĩa bằng 2



Xét bài toán tháp Hà nội với n = 2 (hình II.3.2). Hàm f0(n) = g0(n) + h0(n), với g0(n) là số đĩa đã di chuyển, h0(n) là thông tin về mối liên hệ giữa n và trạng thái đích, chẳng hạn:

- Nếu ở cọc C có 2 đĩa, đĩa nhỏ trên đĩa to thì h0 = 0.

- Nếu ở cọc C có 1 đĩa to thì h0 = 1.

- Nếu ở cọc C chưa có đĩa nào thì h0 = 2.

- Nếu ở cọc C có 1 đĩa nhỏ thì h0 = 3

Lời giải tối ưu cho bài toán tháp Hà Nội được trình bày qua cây lời giải in đậm trong hình II.3.2.

## Câu 3. Giả sử ta đã xét các đỉnh 1, 2, …, 6 bằng chiến lược tìm kiếm theo chiều rộng. Hãy thực hiện từng bước việc kiểm tra tính giải được của đỉnh 1 trên cây Và/Hoặc sau:

A picture containing object, clock

Description automatically generated

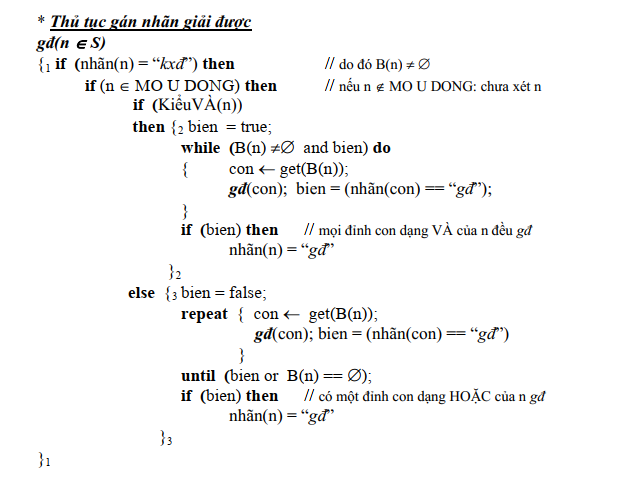
Với mỗi đỉnh n ∈ S, ta dùng các ký hiệu:

- Nếu n là đỉnh kết thúc thì kt(n) = true, ngược lại kt(n) = false.

- Nhãn(n) =

* “gđ” , khi n là đỉnh giải được.
* “kgđ” , khi n là đỉnh không giải được.
* “kxđ” , (không xác định) nếu n chưa đủ thông tin để quyết định hoặc n chưa được xét tới.

- KiểuVÀ(n)=true nếu các đỉnh con của n là đỉnh VÀ và nhận trị false nếu ngược lại.



## Câu 4. Kiểm tra tính hằng đúng của các biểu thức sau:

1. bằng thuật toán Vương Hạo
2. bằng thuật toán Robinson
3. bằng cả hai thuật toán Vương Hạo và Robinson
4. bằng thuật toán suy diễn lùi.

# ĐỀ THI LỚP CTK38

## Câu 1. Hãy tìm đường đi ngắn nhất từ “A” đến “I” hoặc “E” trên cây sau:

A clock in the middle of a watch

Description automatically generated

Bằng các chiến lược (minh họa từng bước thực hiện qua các tập Đóng, Mở)

1. Tìm kiếm sâu dần với giới hạn độ sâu K = 2
2. Tìm kiếm cực tiểu. Trong đó, chi phí đi giữa hai đỉnh cho bởi các số in nghiêng (ví dụ chi phí đi từ A đến B là *15*, từ B đến G là *3*)

Tìm kiếm sâu dần:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tập mở** | **Đỉnh x** | **Độ sâu của x** | **Con của x** | **Tập đóng** |
|  | A | 0 | C, B | Rỗng |
| C, B | C | 1 | F, I | A |
| F, I, B | F | 2 | L, K | A, C |
| I, B, L, K | I (Đích) | 2 | Rỗng | A, C, F |
| B, L, K | B | 1 | G, D | A, C, F, I |

Đường đi ngắn nhất bằng chiến lược tìm kiếm sâu dần với k = 2 là: A, C, F, I.

Tìm kiếm cực tiểu:

*(Lưu ý, tập mở phải sắp xếp theo chi phí tăng dần)*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tập mở** | **Đỉnh x** | **Con của x** | **Tập đóng** |
|  | **A(0)** | C(10), B(15) | Rỗng |
| C(10), B(15) | C(10) | F(15), I(30) | A(0) |
| B(15), F(15), I(30) | **B(15)** | G(3), D(15) | A(0), C(10) |
| G(3), D(15), F(15), I(30) | G(3) | Rỗng | A(0), C(10), B(15) |
| D(15), F(15), I(30) | **D(15)** | E(5), U(15) | A(0), C(10), B(15), G(3) |
| E(5), F(15), U(15), I(30) | **E(5) (Đích)** | Rỗng | A(0), C(10), B(15), G(3), D(15) |
| F(15), U(15), I(30) | F(15) | L(5), K(7) | A(0), C(10), B(15), G(3), D(15), E(5) |

Đường đi ngắn nhất bằng chiến lược AT là: A, B, D, E với chi phí là 35.

## Câu 2. Xét bài toán đẩy số Tocci với n = 3 có hình trạng đầu x0 và hình trạng đích x\*, như sau:

A picture containing object, clock

Description automatically generated

Hàm heuristic h0(x) là số lượng chữ số (khác trống) trong trạng thái x không nằm ở đúng vị trí của chúng trong trạng thái đích x\*. Hãy minh họa từng bước áp dụng thuật giải AKT (tìm kiếm cực tiểu với tri thức heuristic bổ sung) để tìm dãy các bước đưa x0 và x\*.

## Câu 3. Kiểm tra tính hằng đúng của các biểu thức sau:

1. bằng thuật toán Vương Hạo
2. bằng thuật toán Robinson

## Câu 4. Áp dụng thuật toán suy diễn lùi kiểm tra rằng từ tập GT = {a, b} có thể suy ra KL = {g} qua tập RULE = {r1: b 🡪 d, r2: d ˄ a 🡪 e, r3: b ˄ c🡪 f, r4: a 🡪 c, r5: e ˄ f 🡪 g, r6: a 🡪 e} hay không?