

TRÍ TUỆ NHÂN TẠO ARTIFICIAL INTELLIGENCE



Chương 2: CÁC PHƯƠNG PHÁP GIẢI QUYẾT VẤN ĐỀ

GVHD: TS. PHAN ANH CANG



Nội dung

- Chương 1: Tổng quan về trí tuệ nhân tạo
- Chương 2: Các phương pháp giải quyết vấn đề
- Chương 3: Các phương pháp biểu diễn tri thức
- Chương 4: Hệ chuyên gia dựa trên luật
- Chương 5: Máy học





Chương 2

CÁC PHƯƠNG PHÁP GIẢI QUYẾT VẤN ĐỀ



Nội dung

2.1 Không gian trạng thái

2.2 Tìm kiếm trên không gian trạng thái

2.3 Tìm kiếm Heuristic



Không gian trạng thái

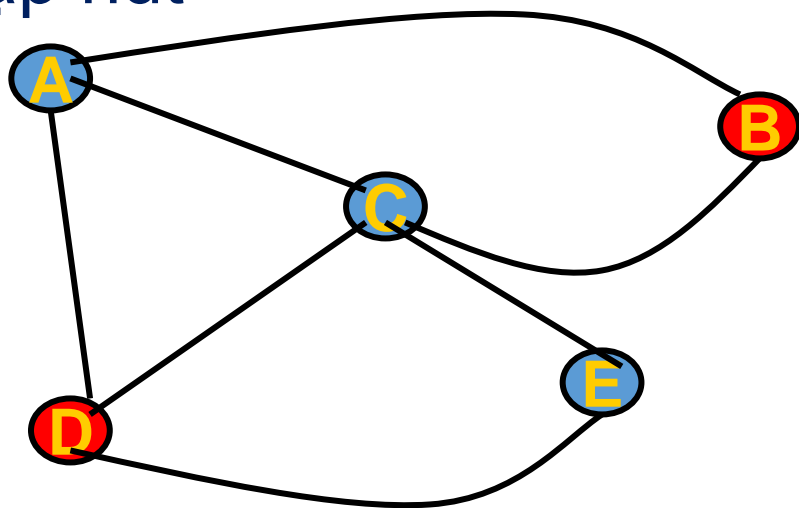
Không gian trạng thái $[N, A, S, GD]$, trong đó:

- N (Node) là tập hợp các nút hay các trạng thái của đồ thị.
- A (Arc) là tập các cung (hay các liên kết) giữa các nút.
- S (Start) là một tập con không rỗng của N , chứa (các) trạng thái ban đầu của bài toán.
- GD (Goal Description) là một tập con không rỗng của N , chứa (các) trạng thái đích của bài toán



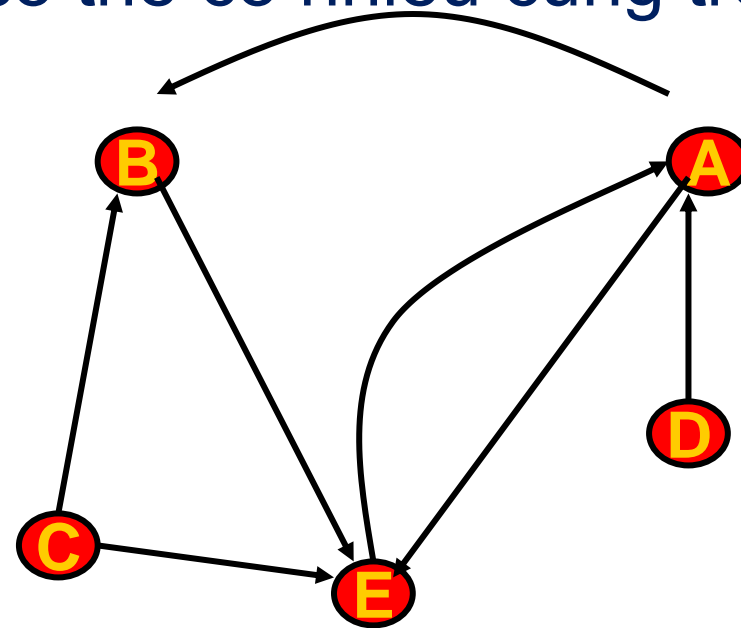
Lý thuyết đồ thị - Review

- Đồ thị: là một cấu trúc bao gồm:
 - Tập các nút $N1, N2, \dots, Nn, \dots$ Không hạn chế
 - Tập các cung nối các cặp nút, có thể có nhiều cung trên một cặp nút



Nút: $\{A, B, C, D, E\}$

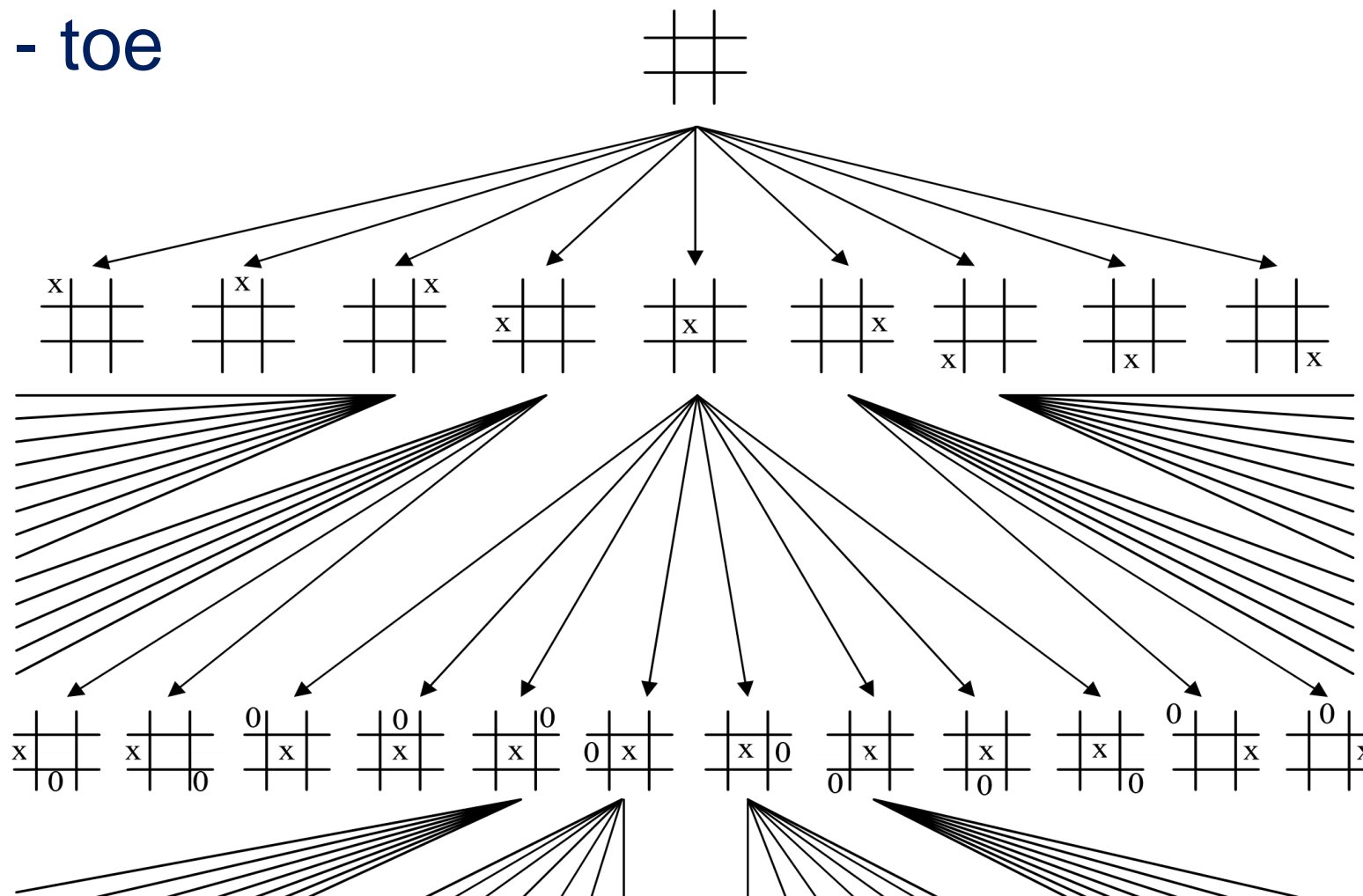
Cung: $\{(a,d), (a,b), (a,c), (b,c), (c,d), (c$





Không gian trạng thái

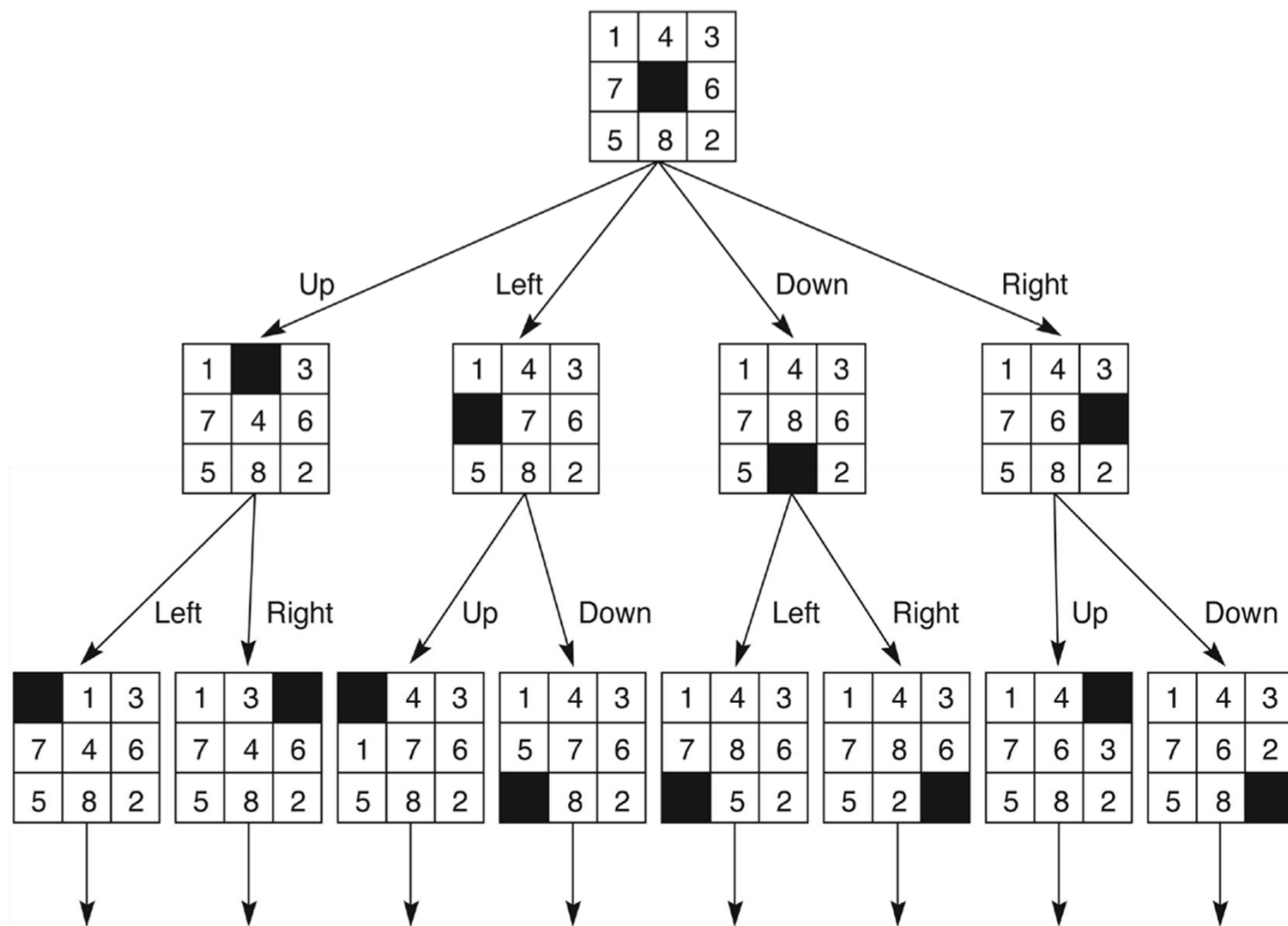
Tic – tac - toe





Không gian bài toán

- Trò 8 ô





Các chiến lược tìm kiếm

2. Tìm kiếm trên không gian trạng thái

2.1. Tìm kiếm hướng từ dữ liệu và tìm kiếm hướng từ mục tiêu

Một không gian trạng thái có thể được tìm kiếm theo hai hướng: từ **các dữ liệu cho trước của một bài toán hướng đến mục tiêu** hay từ **mục tiêu hướng ngược về các dữ liệu**.

*a. Tìm kiếm hướng từ dữ liệu (data-driven search) hay còn gọi là **suy diễn tiến (forward chaining)***: tiến trình giải bài toán bắt đầu với các sự kiện cho trước của bài toán và một tập các luật hợp thức dùng thay đổi trạng thái. Quá trình tìm kiếm được thực hiện bằng cách áp dụng các luật vào các sự kiện để tạo ra các sự kiện mới, sau đó các sự kiện mới này lại được áp dụng các luật để sinh ra các sự kiện mới hơn cho đến khi chúng có thể đưa ra một giải pháp thỏa mãn điều kiện mục tiêu



Các chiến lược tìm kiếm

2. Tìm kiếm trên không gian trạng thái

2.1. Tìm kiếm hướng từ dữ liệu và tìm kiếm hướng từ mục tiêu

b. *Tìm kiếm hướng từ mục tiêu (goal-driven search) hay suy diễn lùi (backward chaining)*

bắt đầu từ mục tiêu mà chúng ta muốn giải quyết, khảo sát xem có thể dùng những luật nào để đạt đến mục tiêu này, đồng thời xác định xem các điều kiện nào phải được thỏa mãn để có thể áp dụng được chúng. Các điều kiện này sẽ trở nên những mục tiêu mới, còn gọi là các mục tiêu phụ (subgoals) trong tiến trình tìm kiếm. Quá trình tìm kiếm cứ tiếp tục như thế, hoạt động theo chiều ngược, cho đến khi gặp các sự kiện thực của bài toán



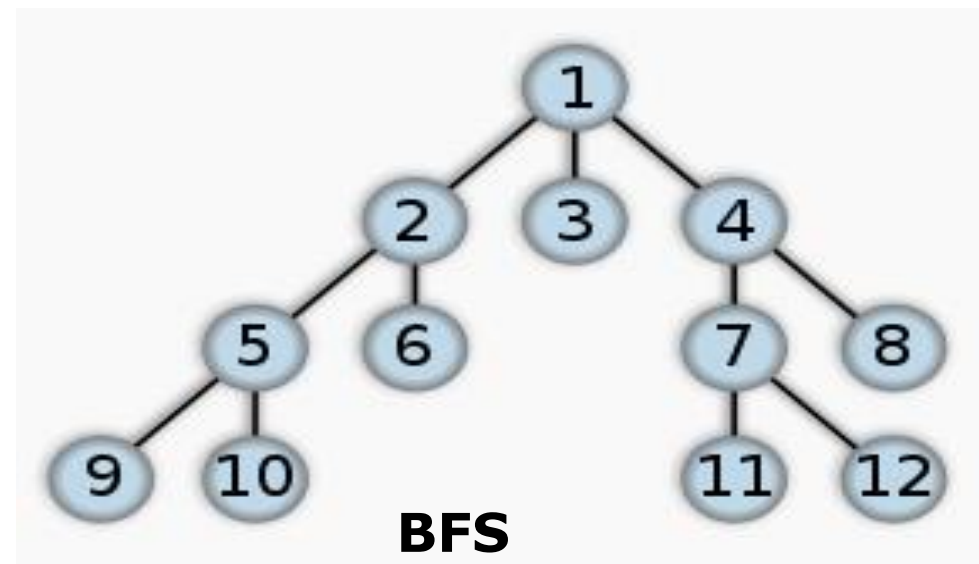
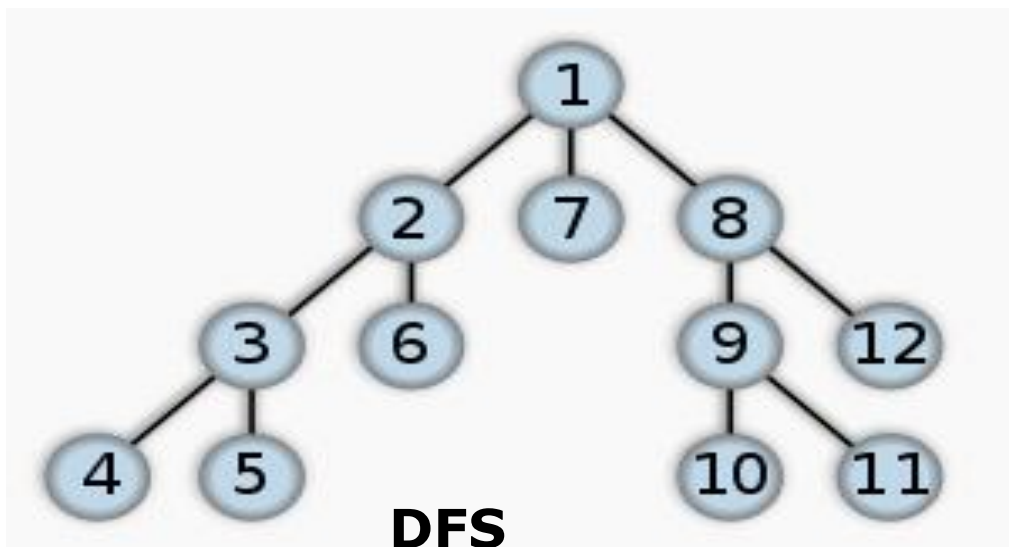
Các chiến lược tìm kiếm

2. Tìm kiếm trên không gian trạng thái

2.2. Tìm kiếm trên đồ thị

Hai cách tìm kiếm phổ biến được áp dụng:

- a. Tìm kiếm theo chiều sâu (Depth First Search - DFS)
- b. Tìm kiếm theo chiều rộng (Breadth First Search - BFS)





Các chiến lược tìm kiếm

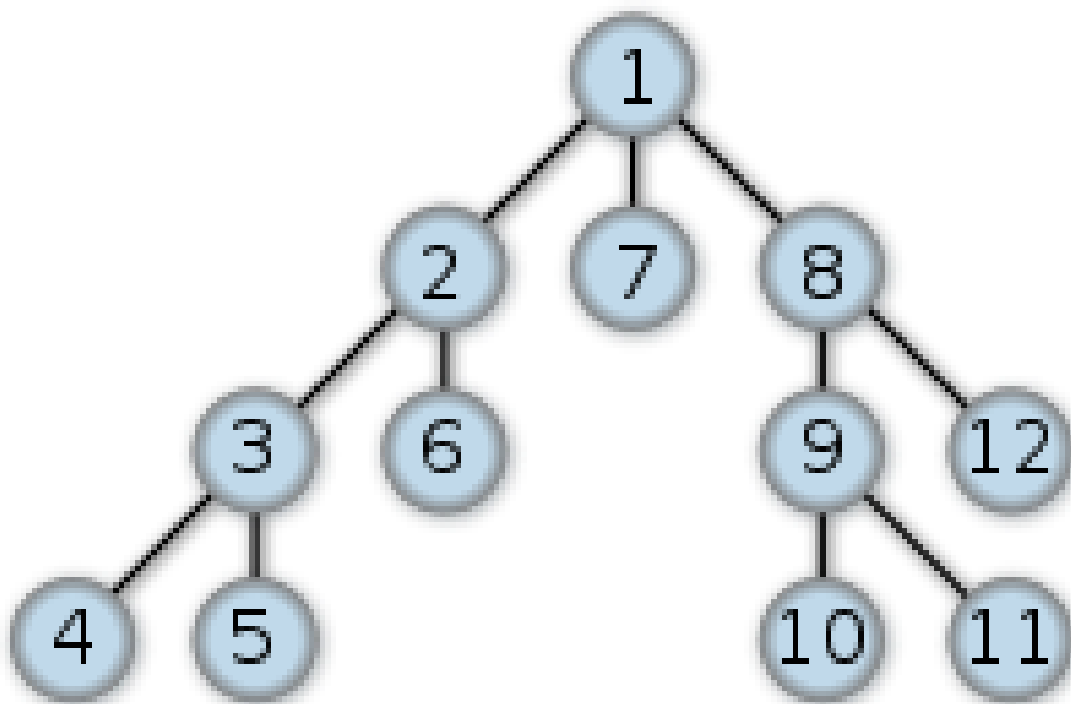
2.1 Tìm kiếm theo chiều sâu

Xuất từ một đỉnh v bất kỳ của đồ thị G , chúng ta thực hiện như sau:

Bước 1: đánh dấu v đã được duyệt.

Bước 2: thực hiện đánh dấu đã duyệt với mỗi đỉnh w chưa duyệt kề với v ,

Bước 3: làm lại bước 2 cho đến khi tất cả các đỉnh được duyệt.



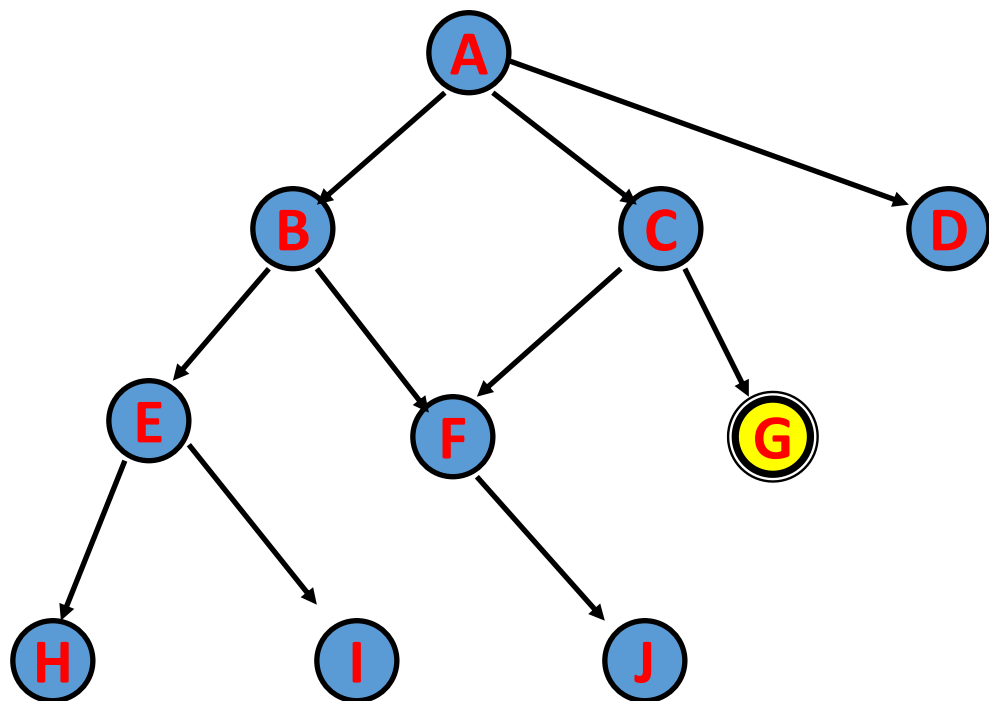
Các nút “con cháu” của nút hiện thời được xem xét trước các nút “anh em”.



Các chiến lược tìm kiếm

2.1. Tìm kiếm theo chiều sâu

Ví dụ: Thực hiện duyệt đồ thị theo chiều sâu trên đồ thị G dưới đây:



Bước	Duyệt	Các nút kề	Nút Đã duyệt
0		[A]	[]
1	A	[B C D]	[A]
2	B	[E F C D]	[A B]
3	E	[H I F C D]	[A B E]
4	H	[I F C D]	[A B E H]
5	I	[F C D]	[A B E H I]
6	F	[J C D]	[A B E H I F]
7	J	[C D]	[A B E H I F J]
8	C	[G D]	[A B E H I F J C]
9	G		

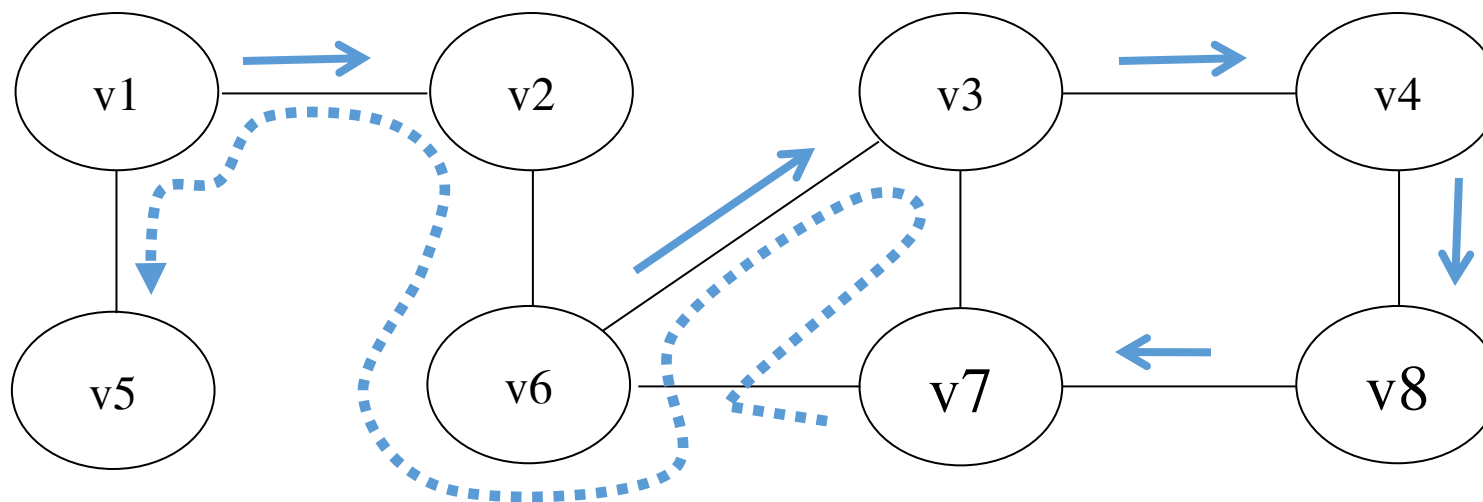


Các chiến lược tìm kiếm

2.1. Tìm kiếm theo chiều sâu

Ví dụ:

Thực hiện duyệt đồ thị theo chiều sâu trên đồ thị G dưới đây:



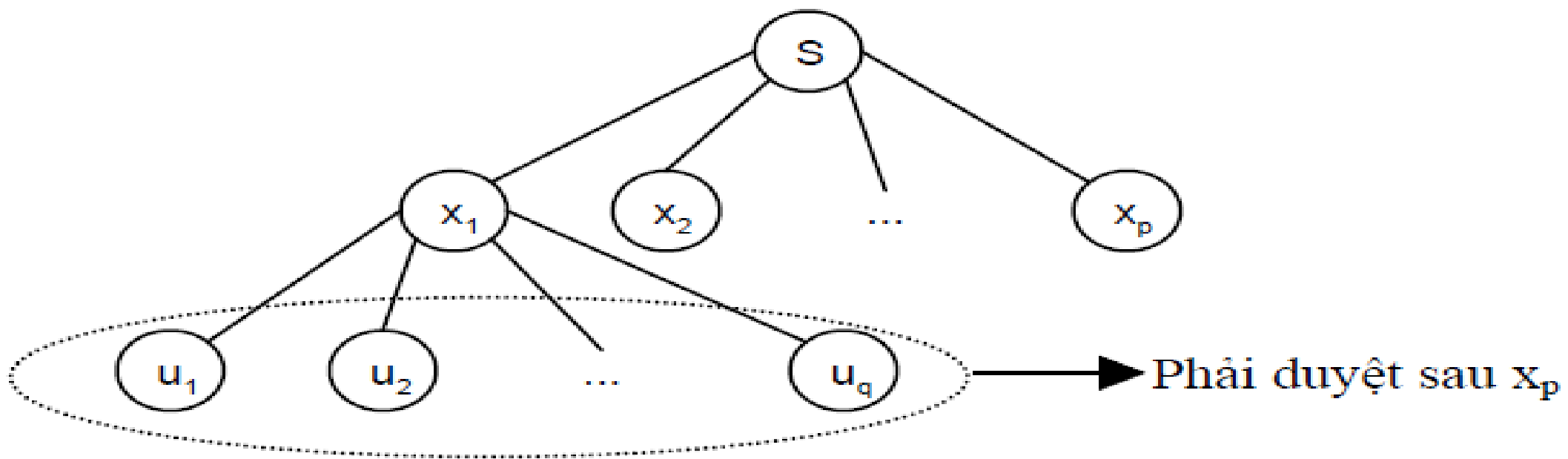
Bảng duyệt

Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
v1	v2	v6	v3	v4	v8	v7	v5



Các chiến lược tìm kiếm

2.2. Tìm kiếm theo chiều rộng





Các chiến lược tìm kiếm

2.2. Tìm kiếm theo chiều rộng

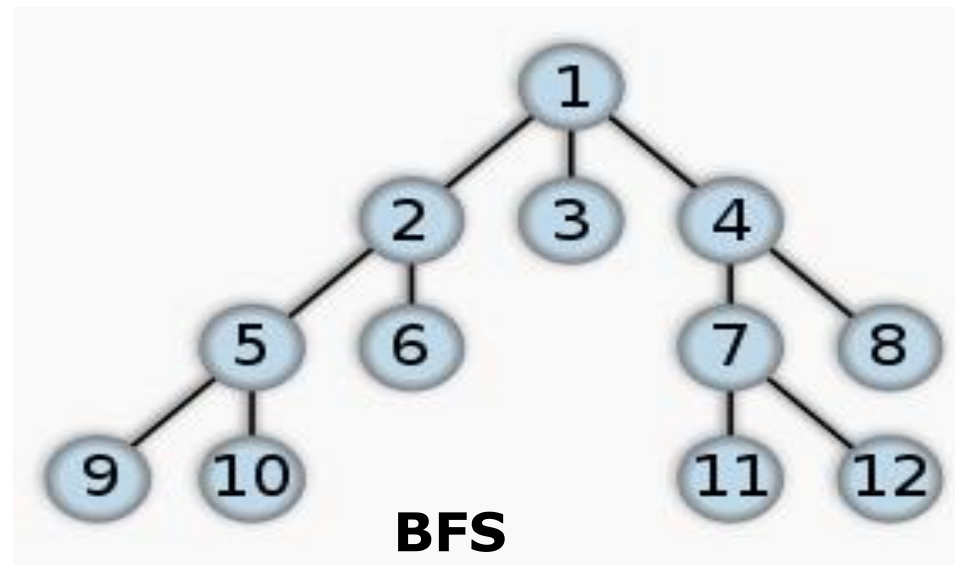
Xuất từ một đỉnh v bất kỳ của đồ thị G , chúng ta thực hiện như sau:

BƯỚC 1: đánh dấu đã duyệt cho một đỉnh v bất kỳ.

BƯỚC 2: chọn đỉnh v đã được duyệt nhưng có đỉnh kề chưa được duyệt. Việc chọn đỉnh v được xét ưu tiên cho các đỉnh được đánh dấu duyệt sớm.

BƯỚC 3: thực hiện đánh dấu đã duyệt với tất cả các đỉnh w kề với v ,

BƯỚC 4: làm lại bước 2 cho đến khi tất cả các đỉnh được duyệt.

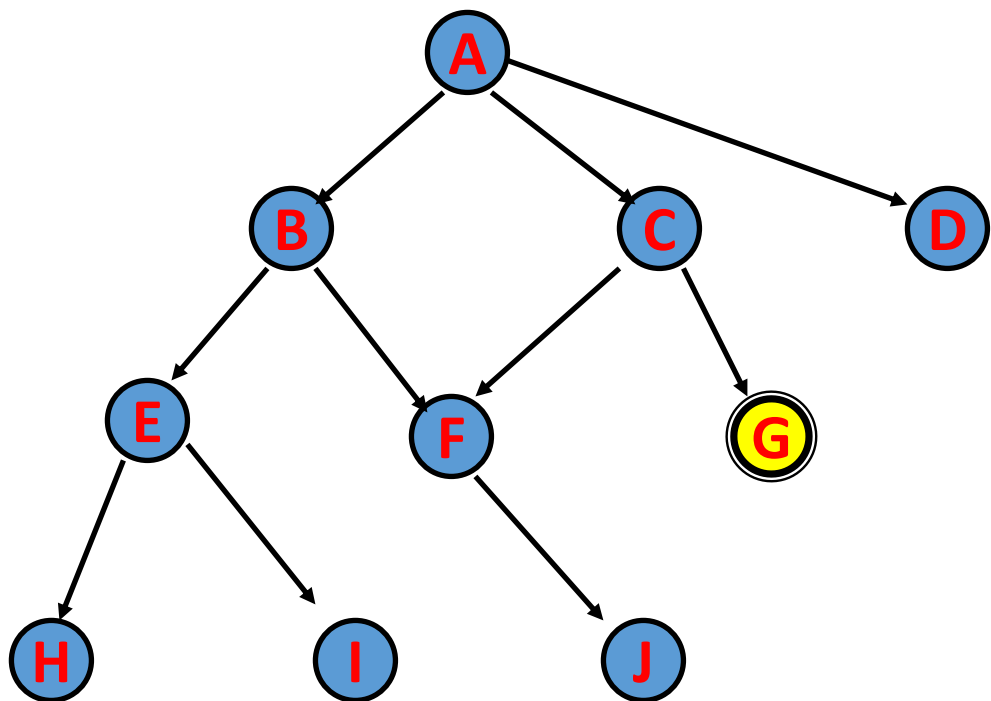




Các chiến lược tìm kiếm

2.2. Tìm kiếm theo chiều rộng

Các nút “anh em” của nút hiện thời được xem xét trước các nút “con cháu”

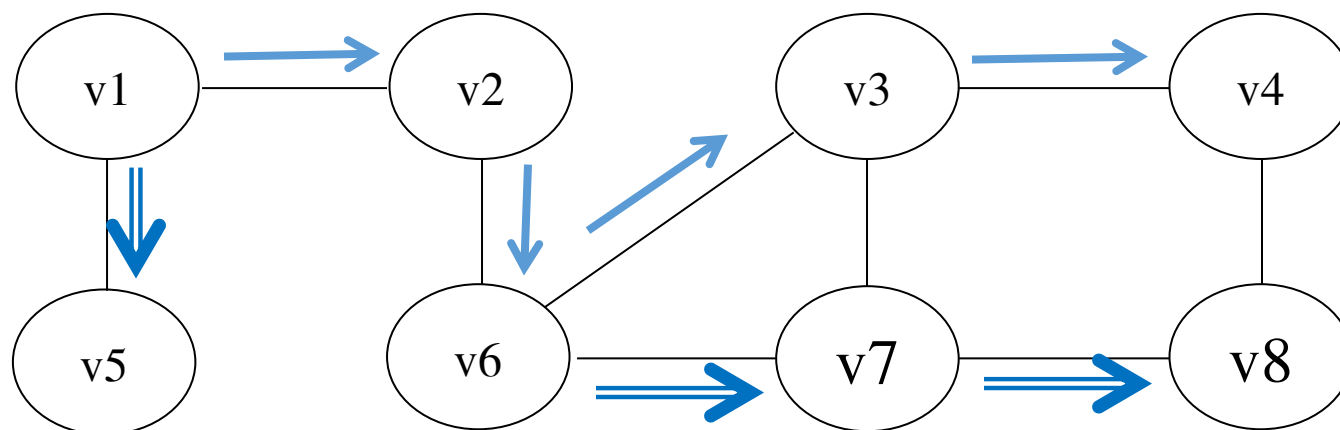


Bước	Duyệt	Các nút kề	Nút Đã duyệt
0		[A]	[]
1	A	[B C D]	[A]
2	B	[C D E F]	[A B]
3	C	[D E F G]	[A B C]
4	D	[E F G]	[A B C D]
5	E	[F G H I]	[A B C D E]
6	F	[G H I J]	[A B C D E F]
7	G	[H I J]	[A B C D E F]



Các chiến lược tìm kiếm

2.2. Tìm kiếm theo chiều rộng



Bảng duyệt

Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
v1	v2	v5	v6	v3	v7	v4	v8



Các chiến lược tìm kiếm

- Breath First: được tổ chức dạng FIFO
- Depth First: được tổ chức dạng LIFO
- Hiệu quả:
 - Breath First luôn tìm ra nghiệm có số cung nhỏ nhất
 - Depth First “thường” cho kết quả nhanh hơn.
- Kết quả
 - Breath First search chắc chắn tìm ra kết quả nếu có.
 - Depth First có thể bị lặp vô tận. Tại sao?????
- Bùng nổ tổ hợp là khó khăn lớn nhất cho các giải thuật này.



Các chiến lược tìm kiếm

- Depth first search có khả năng lặp vô tận do các trạng thái con sinh ra liên tục. Độ sâu tăng vô tận.
- Khắc phục bằng cách giới hạn độ sâu của giải thuật.
- Sâu bao nhiêu thì vừa?
- Chiến lược giới hạn:
 - Cố định một độ sâu MAX, như các danh thủ chơi cờ tính trước được số nước nhất định
 - Theo cấu hình resource của máy tính
 - Meta knowledge trong việc định giới hạn độ sâu.
- Giới hạn độ sâu => co hẹp không gian trạng thái => có thể mất nghiệm.



3.1. Heuristic là gì?

- **Heuristic là gì?**
 - Heuristic là những tri thức được rút ra từ những kinh nghiệm, “trực giác” của con người.
 - Heuristic có thể là những tri thức “đúng” hay “sai”.
 - Heuristic là những siêu tri thức (meta knowledge) và “thường đúng”.

- **Heuristic dùng để làm gì?**

Sử dụng heuristic trong hai tình huống cơ bản:

- Bài toán được định nghĩa chính xác nhưng *chi phí tìm lời giải bằng TK vét cạn là không thể chấp nhận*.
VD: Sự bùng nổ KGTK trong trò chơi cờ vua.
- *Vấn đề với nhiều sự mơ hồ* trong lời phát biểu bài toán hay dữ liệu cũng như tri thức sẵn có.
VD: Chẩn đoán trong y học.

Heuristic giúp tìm kiếm đạt kết quả với chi phí thấp hơn



3.1. Heuristic là gì?

- Một giải thuật heuristic có thể được xem gồm 2 phần:
 - **Phép đo heuristic:** thể hiện qua *hàm đánh giá heuristic (evaluation function)*, dùng để đánh giá các đặc điểm của một trạng thái trong KGTT.

Hàm đánh giá heuristic :

$$F(n) := G(n) + H(n)$$

G(n): Chi phí thực từ trạng thái n đến trạng thái bắt đầu

H(n): Chi phí ước lượng heuristic từ trạng thái n đến goal.

- **Giải thuật tìm kiếm heuristic:**
 - *TK tốt nhất (best-first search)*
 - *Giải thuật leo núi (hill-climbing)*
 - Giải thuật A*



3.2. Các phương pháp tìm kiếm Heuristic

3.2.1 Tìm kiếm tối ưu

Xét ví dụ là bài toán 8 puzzle với:

2	8	3
1	6	4
7		5

Bắt đầu

1	2	3
8		4
7	6	5

Mục tiêu

♦ Hàm đánh giá Heuristic: $F(n) = G(n) + H(n)$

Với $G(n)$: số lần chuyển vị trí ô trống đã thực hiện

$H(n)$: Số ô chữ số nằm sai vị trí

Nút X có giá trị heuristic tốt hơn nút Y nếu $F(x) < F(y)$.

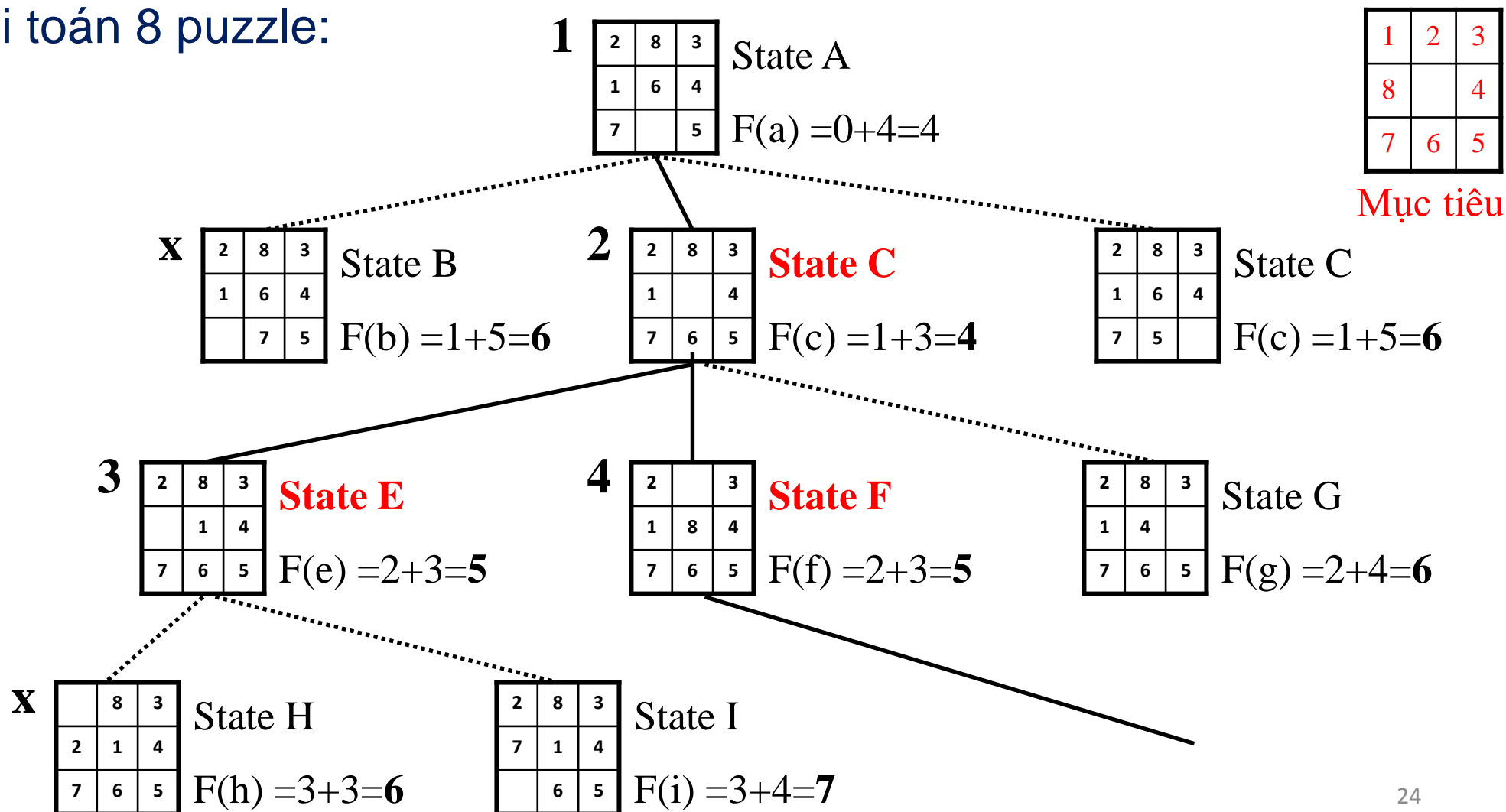
Ta có hoạt động của giải thuật Best First search trên như hình sau:



3.2. Các phương pháp tìm kiếm Heuristic

3.2.1 Tìm kiếm tốt nhất (best-first search)

Xét ví dụ là bài toán 8 puzzle:

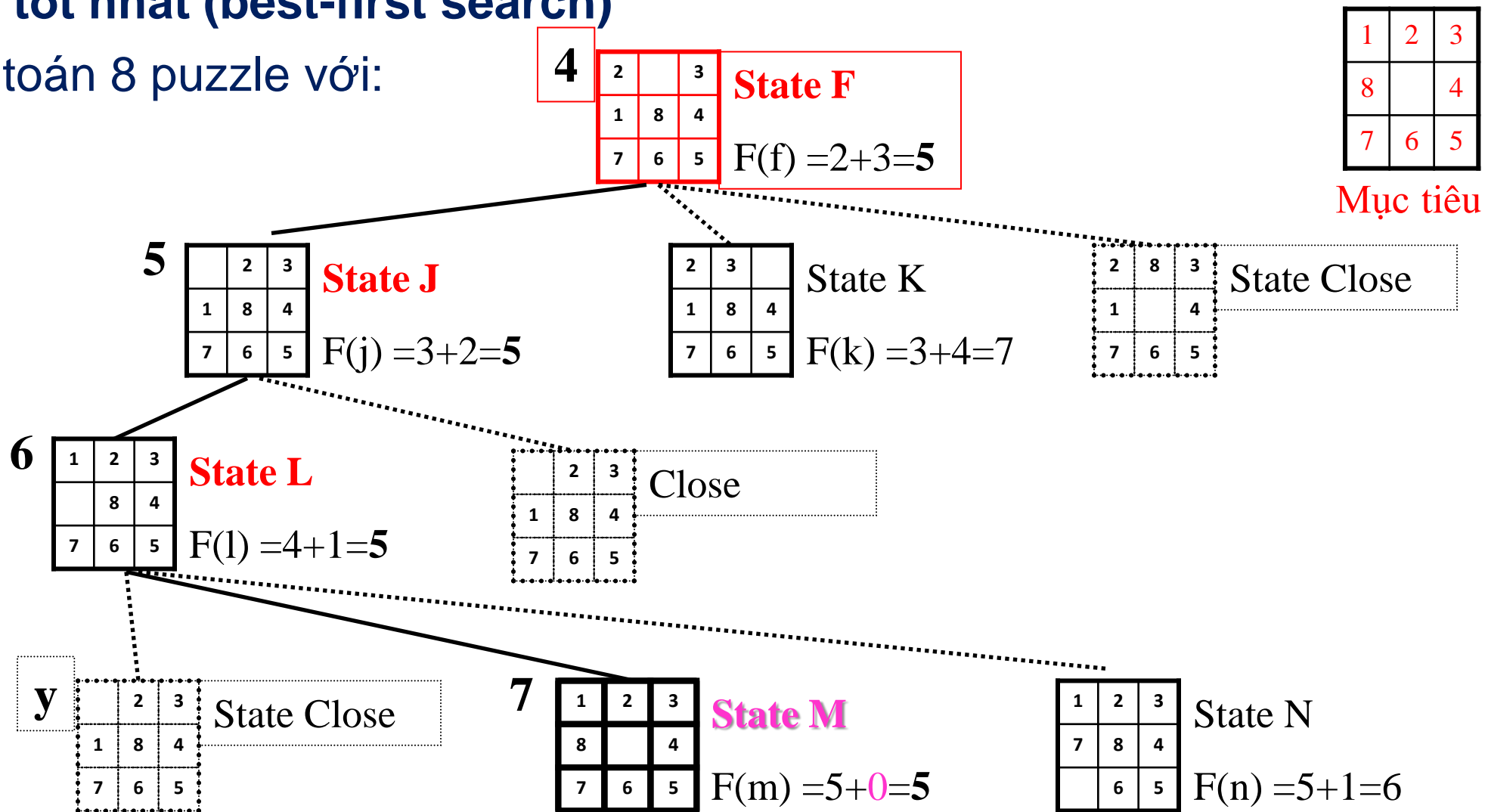




3.2. Các phương pháp tìm kiếm Heuristic

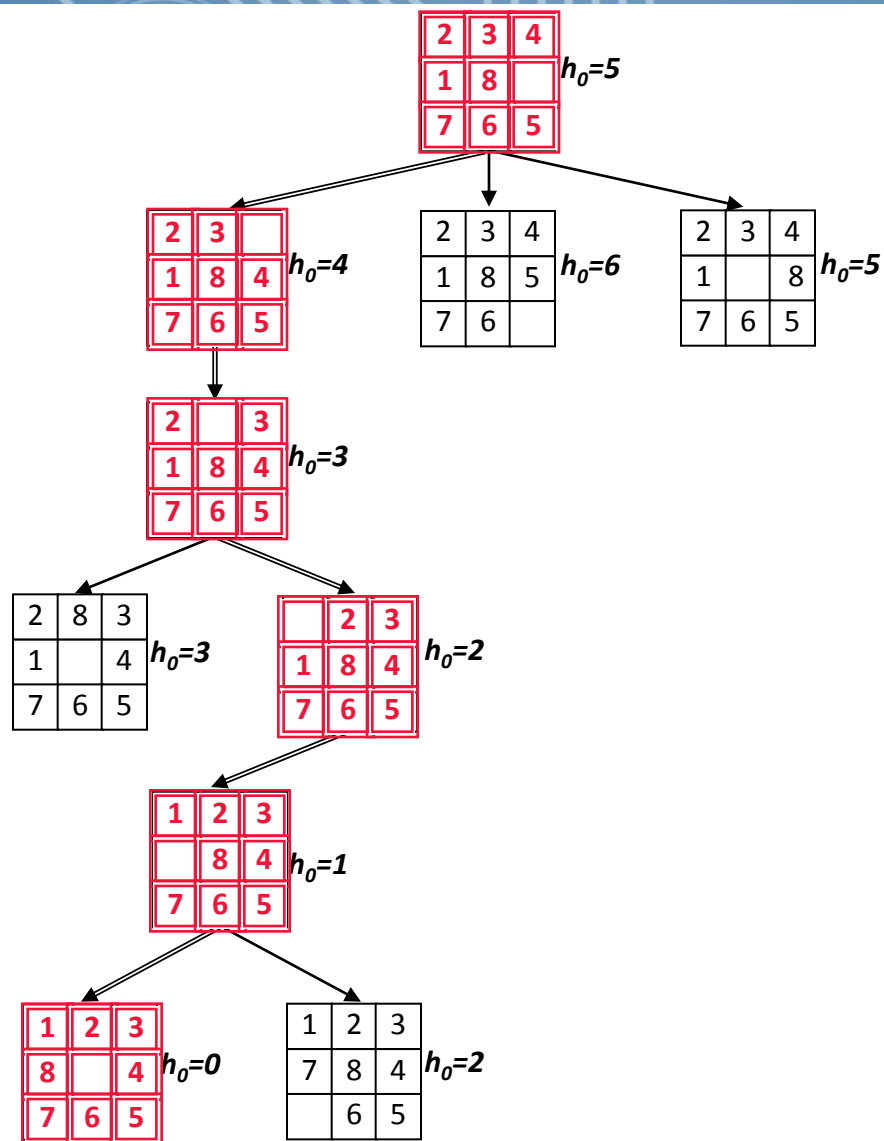
3.2.1 Tìm kiếm tốt nhất (best-first search)

Xét ví dụ là bài toán 8 puzzle với:





Ví dụ





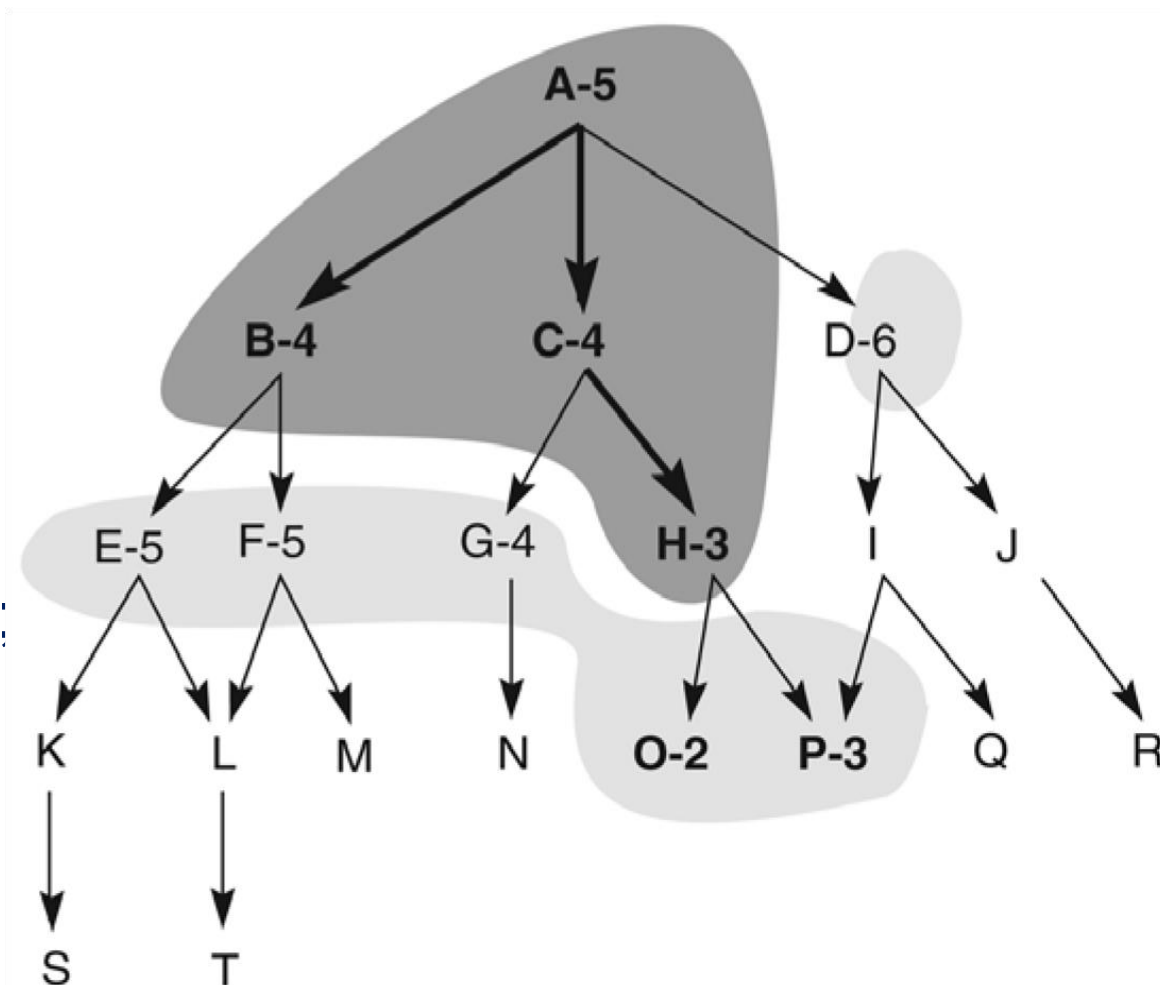
Tìm kiếm ưu tiên tối ưu

- Sử dụng Heuristic
- Chọn trạng thái tốt nhất trong các trạng thái đã sinh ra
- Có “nhìn sang hai bên”



Ví dụ

- Open = [A5]; Closed = []
- A5: Open = [B4,C4,D6];
Closed = [A5]
- B4: Open = [C4,E5,F5,D6];
Closed = [B4,A5]
- C4: Open = [H3,G4,E5,F5,D6];
Closed = [C4,B4,A5]
- H3: Open = [O2,P3,G4,E5,F5,D6];
Closed = [H3,C4,B4,A5]
- O2: Open = [P3,G4,E5,F5,D6];
Closed = [O2,H3,C4,B4,A5]
- P3: Tìm được lời giải!



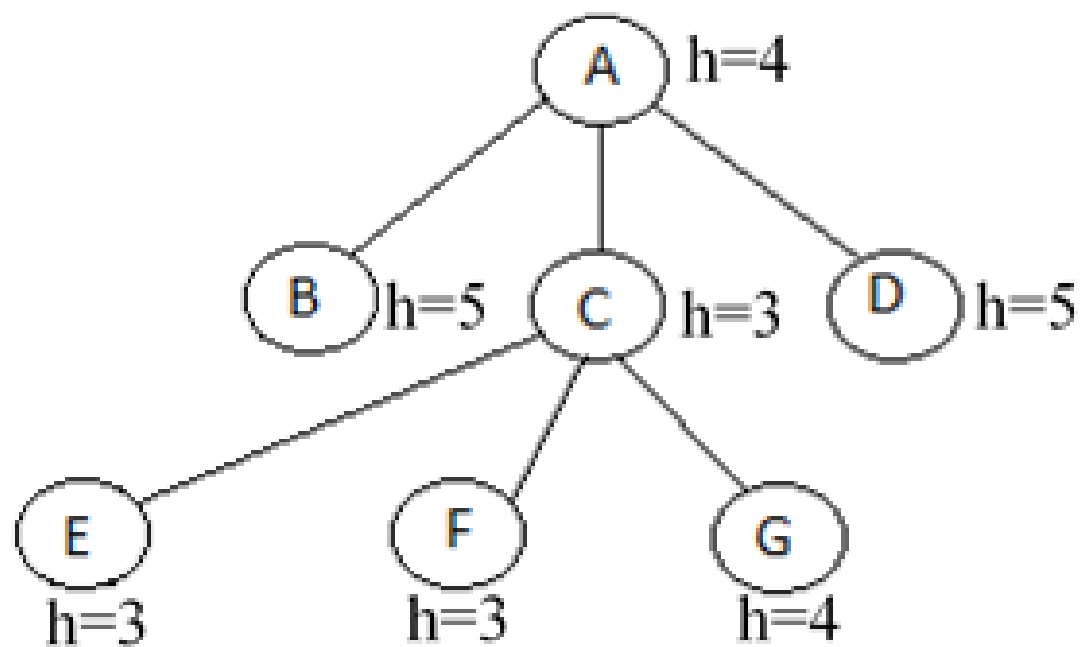


Ví dụ

Closed = [B4, A5];

- C4: Open = [H3, G4, E5, F5, D6];
Closed = [C4, B4, A5]
- H3: Open = [O2, P3, G4, E5, F5, D6];
Closed = [H3, C4, B4, A5]
- O2: Open = [P3, G4, E5, F5, D6];
Closed = [O2, H3, C4, B4, A5]
- P3: Tìm được lời giải!

- Tìm đường đi từ A đến F





Giải thuật A*

- $f(X) = g(X) + h(X)$: phản ánh độ tốt
 - g : giá trị **đường đi thực** từ điểm xuất phát đến X
 - h : **ước lượng đoạn đường còn lại** từ X đến đích
- $prev(X)$: trạng thái ngay trước X thỏa điều kiện đường đi đến X ngắn nhất (tốt nhất) tại thời điểm đang xét
- Nếu $Y = prev(X)$ thì $g(X) = g(Y) + cost(Y, X)$



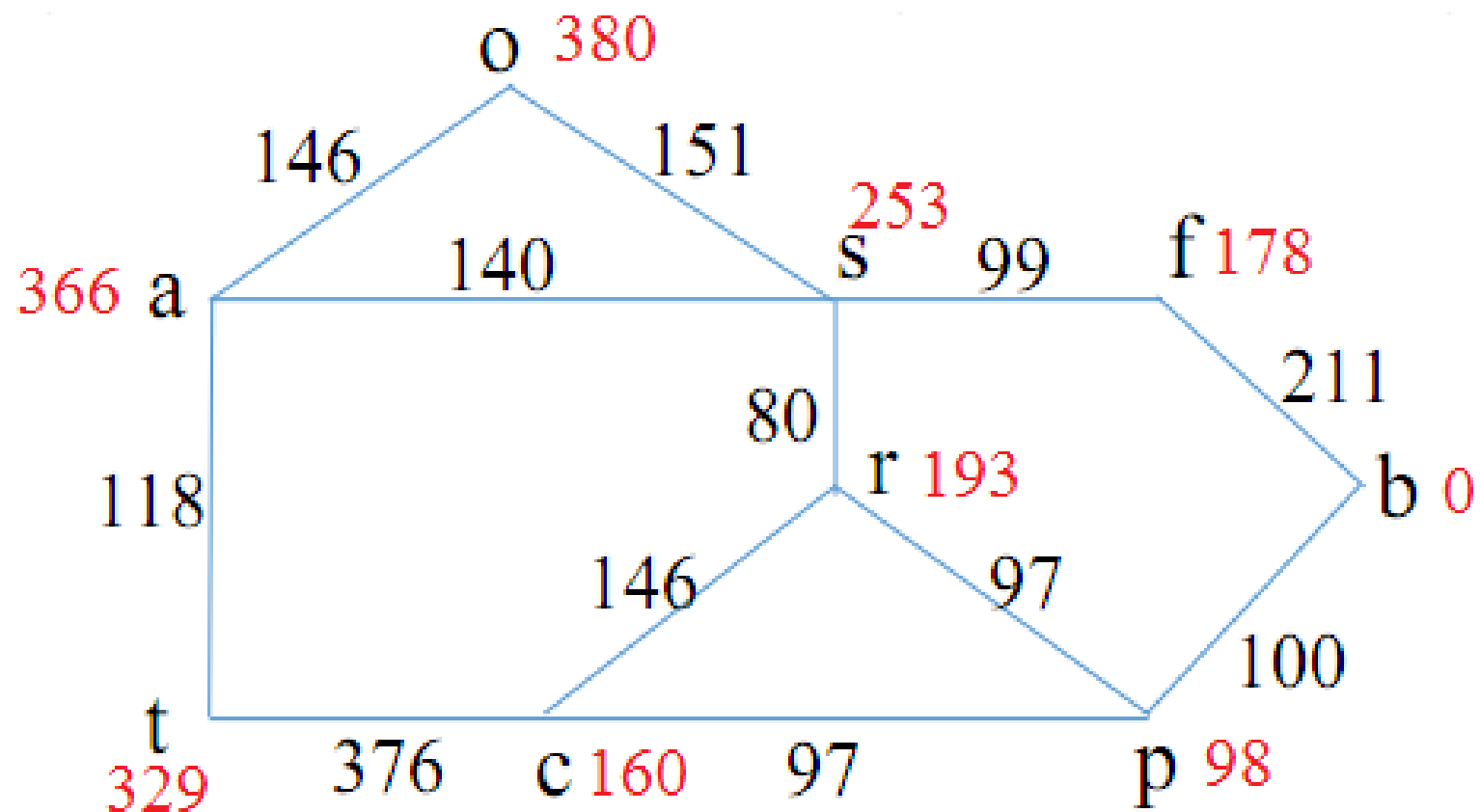
Giải thuật A*

Lưu ý về giải thuật

- Chọn X với điều kiện $f(X)$ nhỏ nhất
- Phát sinh con X_k của X
- Nếu X_k mới **tốt hơn** X_k cũ trong **Open** \rightarrow thay thế:
 - Cập nhật lại $g(X_k) = g(X) + \text{cost}(X, X_k)$
 - Cập nhật lại $f(X_k) = g(X_k) + h(X_k)$
 - Cập nhật lại $\text{prev}(X_k) = X$
- Nếu X_k mới **tốt hơn** X_k cũ trong **Closed** \rightarrow thay thế + lan truyền
- Nếu **chưa có** thì **thêm X_k vào Open**



Giải thuật A*





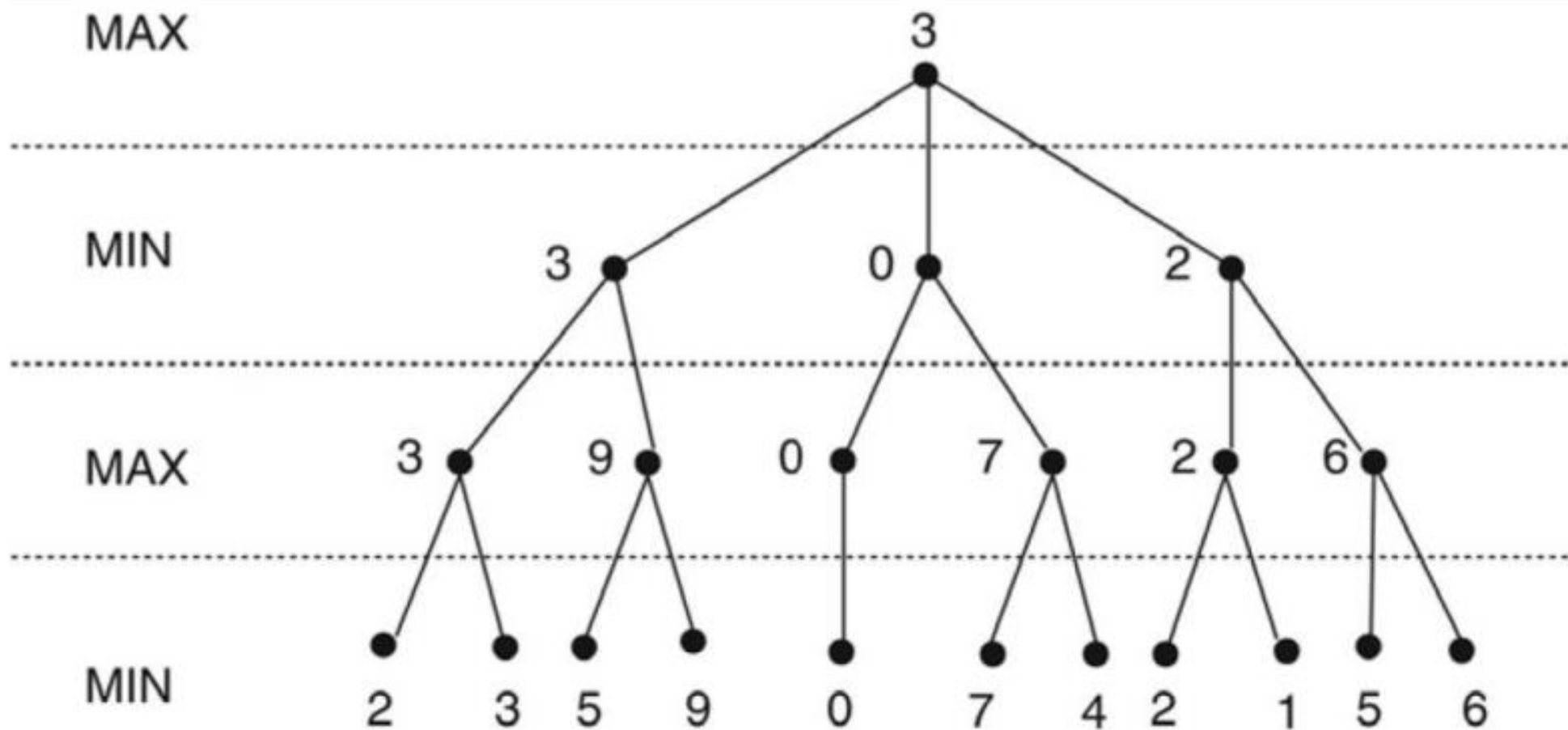
Giải thuật Minimax

Giải thuật minimax:

- Hai đấu thủ trong trò chơi được gọi là MIN và MAX.
- Mỗi nút lá có giá trị:
 - 1 nếu là MAX thắng,
 - 0 nếu là MIN thắng.
- Minimax sẽ truyền các giá trị này lên cao dần trên đồ thị, qua các nút cha mẹ kế tiếp theo các luật sau:
 - Nếu trạng thái cha mẹ là MAX, gán cho nó giá trị **lớn nhất** có trong các trạng thái con.
 - Nếu trạng thái bố, mẹ là MIN, gán cho nó giá trị **nhỏ nhất** có trong các trạng thái con.

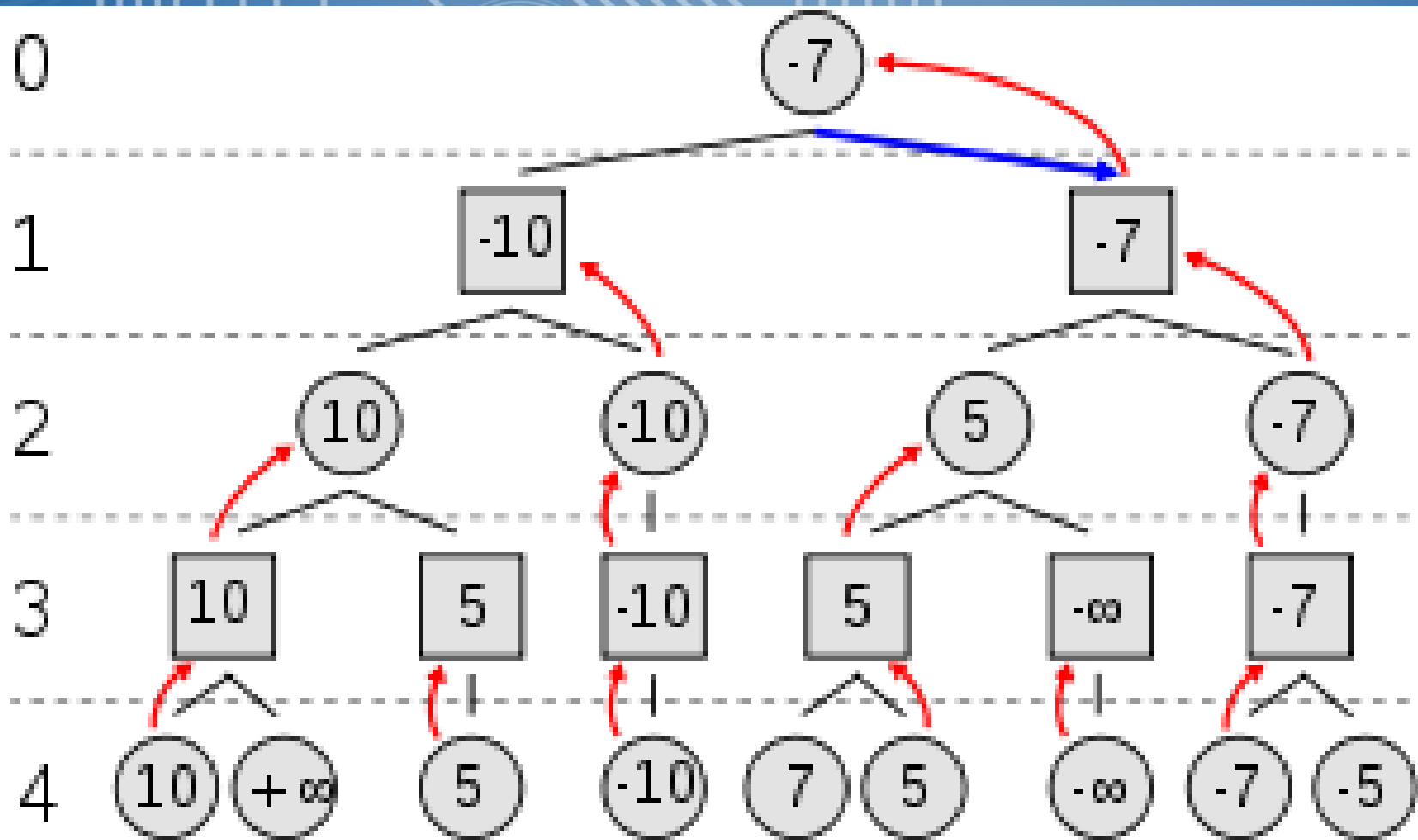


Thủ tục Minimax





Thủ tục Minimax



Mức 0, 2, 4 là lượt đi của máy - mức MAX (kết quả lượng giá càng lớn càng tốt).
Mức 1, 3 là lượt đi của người - mức MIN (kết quả lượng giá càng nhỏ càng tốt).



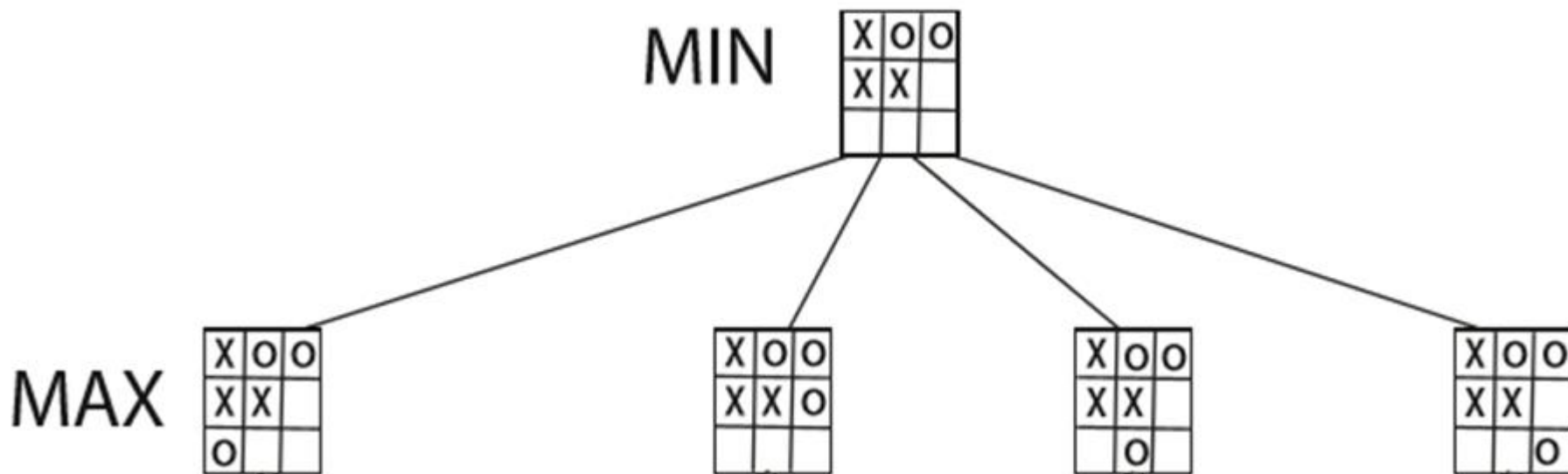
Thủ tục Minimax

Ví dụ mô phỏng giải thuật Minimax cho trò chơi Tic-Tac-toe

MAX đại diện quân đi O.

MIN đại diện quân đi X.

Trạng thái kết thúc là trạng thái có 3 ô liên tiếp ngang, dọc, chéo có cùng một quân cờ X hoặc O, nếu là X tức MIN thắng còn O tức MAX thắng còn nếu tất cả các ô cờ đều được đi và trạng thái. Điểm thắng của X là -1, của O là 1, và bàn cờ hòa là 0.





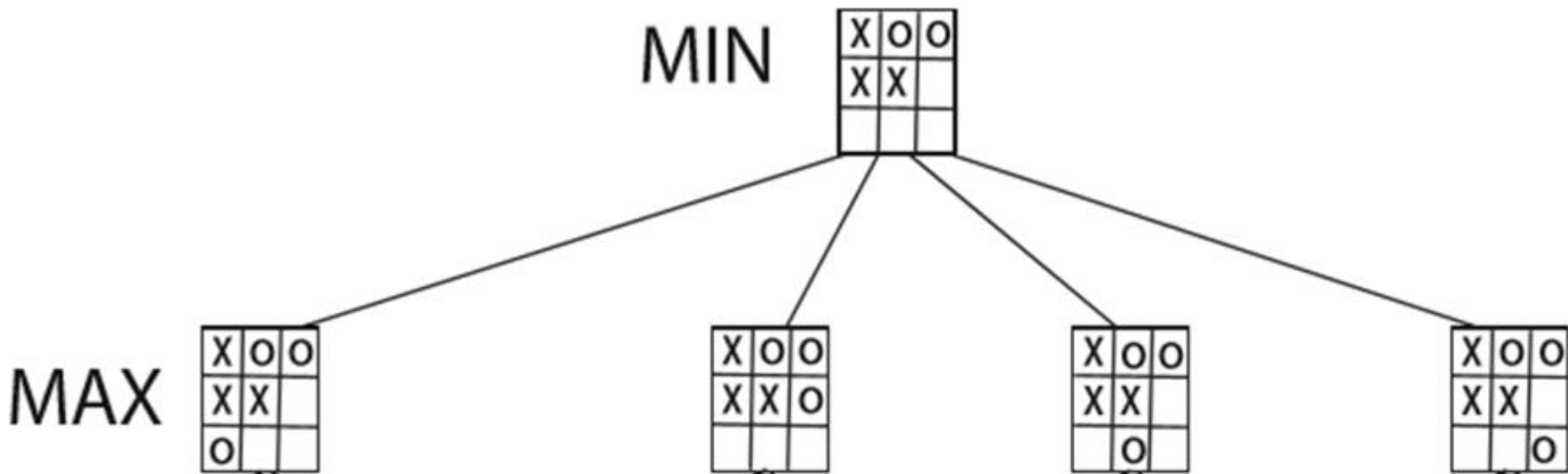
Thủ tục Minimax

Ví dụ mô phỏng giải thuật Minimax cho trò chơi Tic-Tac-toe

Yêu cầu:

Áp dụng giải thuật minmax:

Từ trạng thái bàn cờ hiện tại ta dự đoán nước đi của trạng thái tiếp theo và xác định giá trị cây trò chơi





Thủ tục Minimax

Trò chơi đối kháng

- Trạng thái kết thúc bàn cờ được định trị (1, 0, -1)
 - Max: người **O** đi mong muốn bàn cờ kết thúc lớn nhất - gán trị 1
 - Min: người **X** đi mong muốn bàn cờ kết thúc nhỏ nhất – gán trị -1
 - Hòa: gán trị 0 (không)
- Định trị nút cho biết khả năng chiến thắng của người chơi:
 - $V(\text{Max}) = \text{Max}(\text{các nút con})$
 - $V(\text{Min}) = \text{Min}(\text{các nút con})$



Thủ tục Minimax – Trò chơi đối kháng

B1: Xác định tất cả các khả năng có thể đi của các quân cờ

B2: Xác định giá trị của các nút lá theo quy tắc:

- Max: O đi 1: thang
- Min: X đi -1: thang
- O hòa

B3: Sử dụng thuật toán minimax để xác định các giá trị truyền về từ nút lá đến nút gốc



Thủ tục Minimax

Ví dụ mô phỏng giải thuật Minimax cho trò chơi Tic-Tac-toe

- Nếu ở trạng thái mà ta gặp chiến thắng nếu đó là lượt đi của quân X thì đánh giá điểm trạng thái đó là -1.
- Nếu ở trạng thái ta gặp chiến thắng nếu đó là lượt đi của quân O thì đánh giá điểm trạng thái đó là 1.
- Nếu là hòa thì điểm trạng thái đó là 0.

Sau đó tính ngược lại cây trò chơi theo quy tắc:

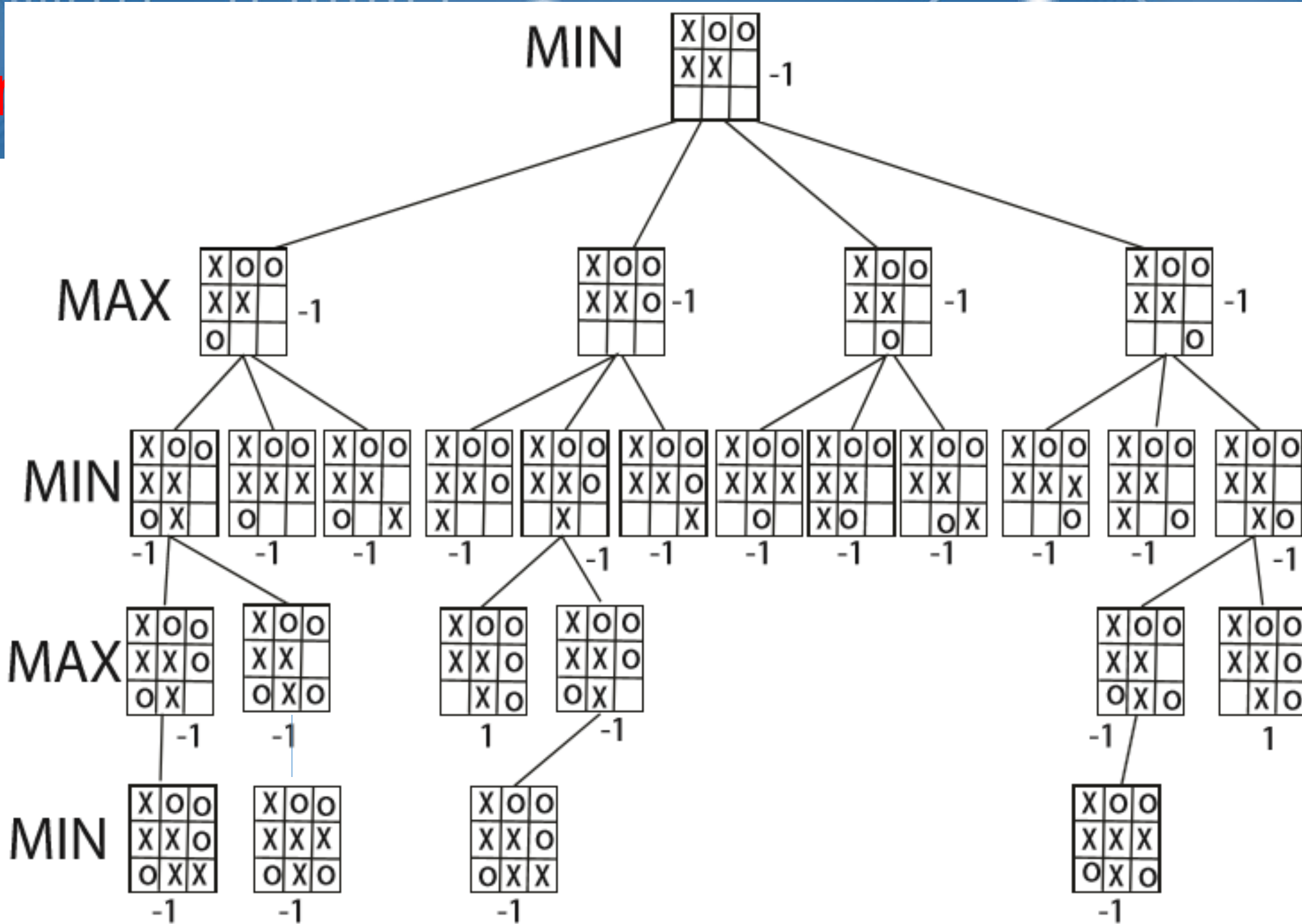
- Nút thuộc lớp MAX thì gán cho nó giá trị lớn nhất của các Node con của Node đó.
- Nút thuộc lớp MIN thì gán cho nó giá trị nhỏ nhất của các Node con của Node đó.

Sau khi lượng giá hết cây trò chơi ta tiến hành chọn bước đi tiếp theo theo nguyên tắc

- Nếu lớp tiếp theo là MAX ta chọn Node con có giá trị lớn nhất.
- Nếu lớp tiếp theo là MIN ta chọn Node con có giá trị nhỏ nhất.



Đi





Thủ tục Minimax

Ưu điểm và khuyết điểm

Ưu điểm

Tìm kiếm được mọi nước đi tiếp theo sau đó lựa chọn nước đi tốt nhất, vì giải thuật có tính chất vét cạn nên không bỏ sót trạng thái.

Khuyết điểm

Đối với các trò chơi có không gian trạng thái lớn như caro, cờ tướng... việc chỉ áp dụng giải thuật Minimax có lẽ không còn hiệu quả nữa sự bùng nổ tổ hợp quá lớn.

Giải thuật áp dụng nguyên lý vét cạn không tận dụng được thông tin của trạng thái hiện tại để lựa chọn nước đi, vì duyệt hết các trạng thái nên tốn thời gian.



Cắt tỉa Alpha - Beta

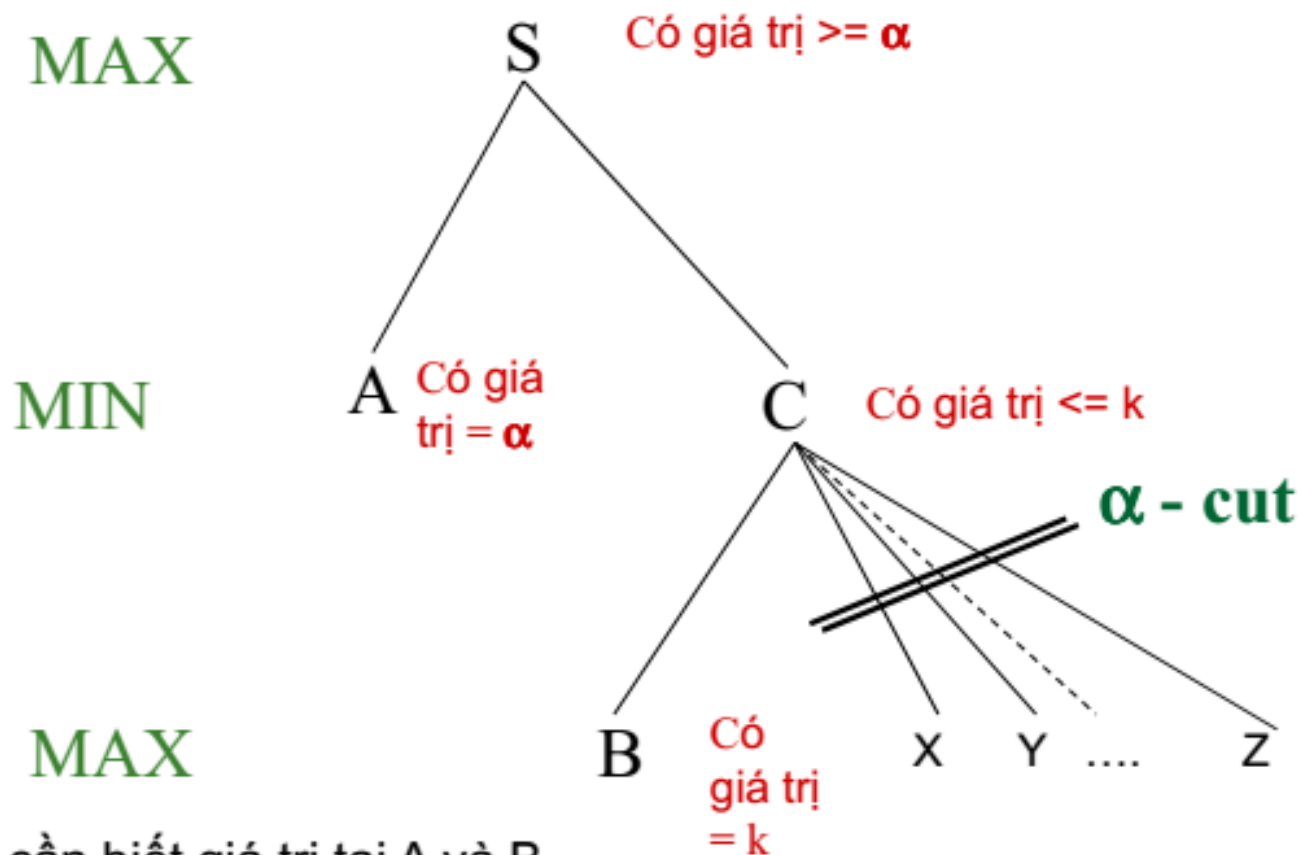
Chiến lược cắt tỉa α - β

- Tìm kiếm theo kiểu depth-first
- Nút MAX có 1 giá trị α (luôn tăng)
- Nút MIN có 1 giá trị β (luôn giảm)
- Tìm kiếm có thể kết thúc dưới bất kỳ
 - Nút MIN nào có $\beta \leq \alpha$ của bất kỳ nút cha MAX nào
 - Nút MAX nào có $\alpha \geq \beta$ của bất kỳ nút cha MIN nào
- Cắt tỉa α - β thể hiện *mối quan hệ giữa các nút ở mức n và $n+2$* , mà tại đó toàn bộ cây có gốc tại mức $n+1$ có thể cắt bỏ



Cắt tỉa Alpha - Beta

Cắt tỉa α (vị trí MAX)



Điều kiện 1: Chỉ cần biết giá trị tại A và B

Điều kiện 2: Giá trị A > giá trị B

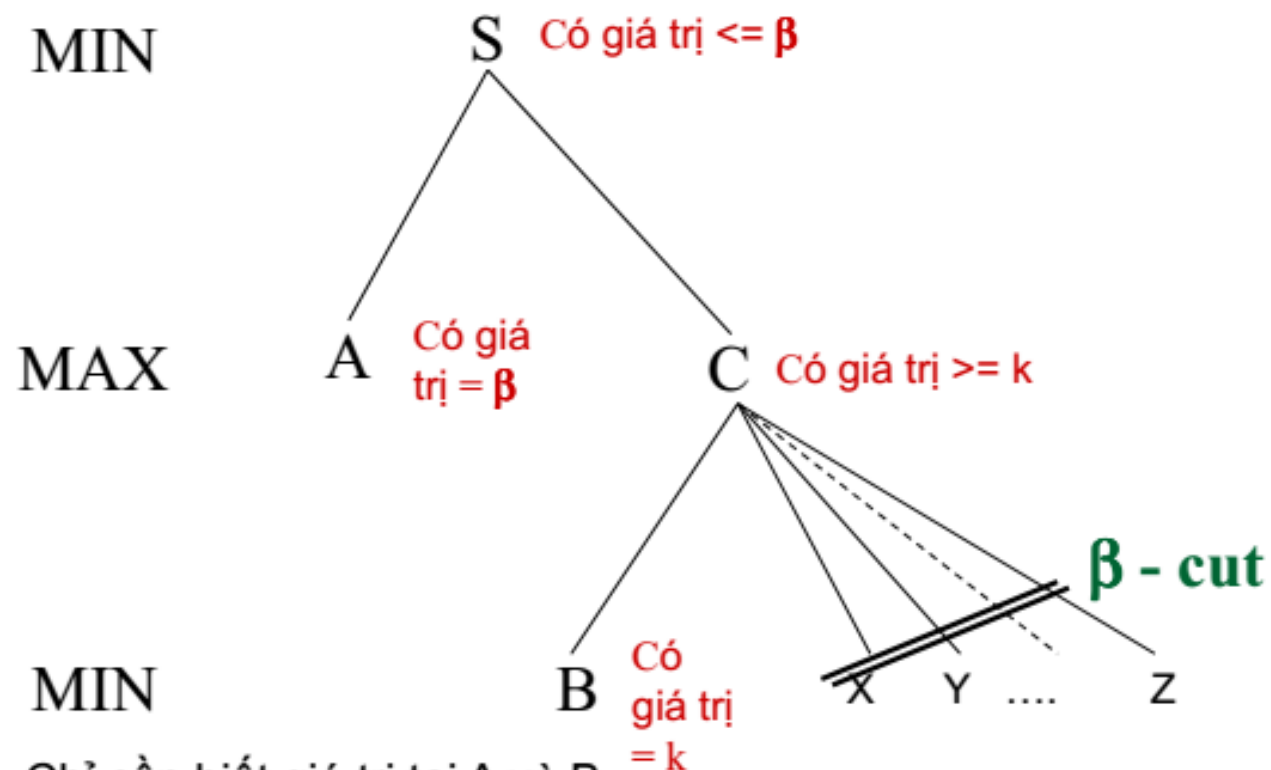
Điều kiện 3: X, Y, .., Z ở vị trí Max

Bỏ những cây con có gốc là X, Y, ..., Z



Cắt tỉa Alpha - Beta

Cắt tỉa β (vị trí Min)



Điều kiện 1: Chỉ cần biết giá trị tại A và B

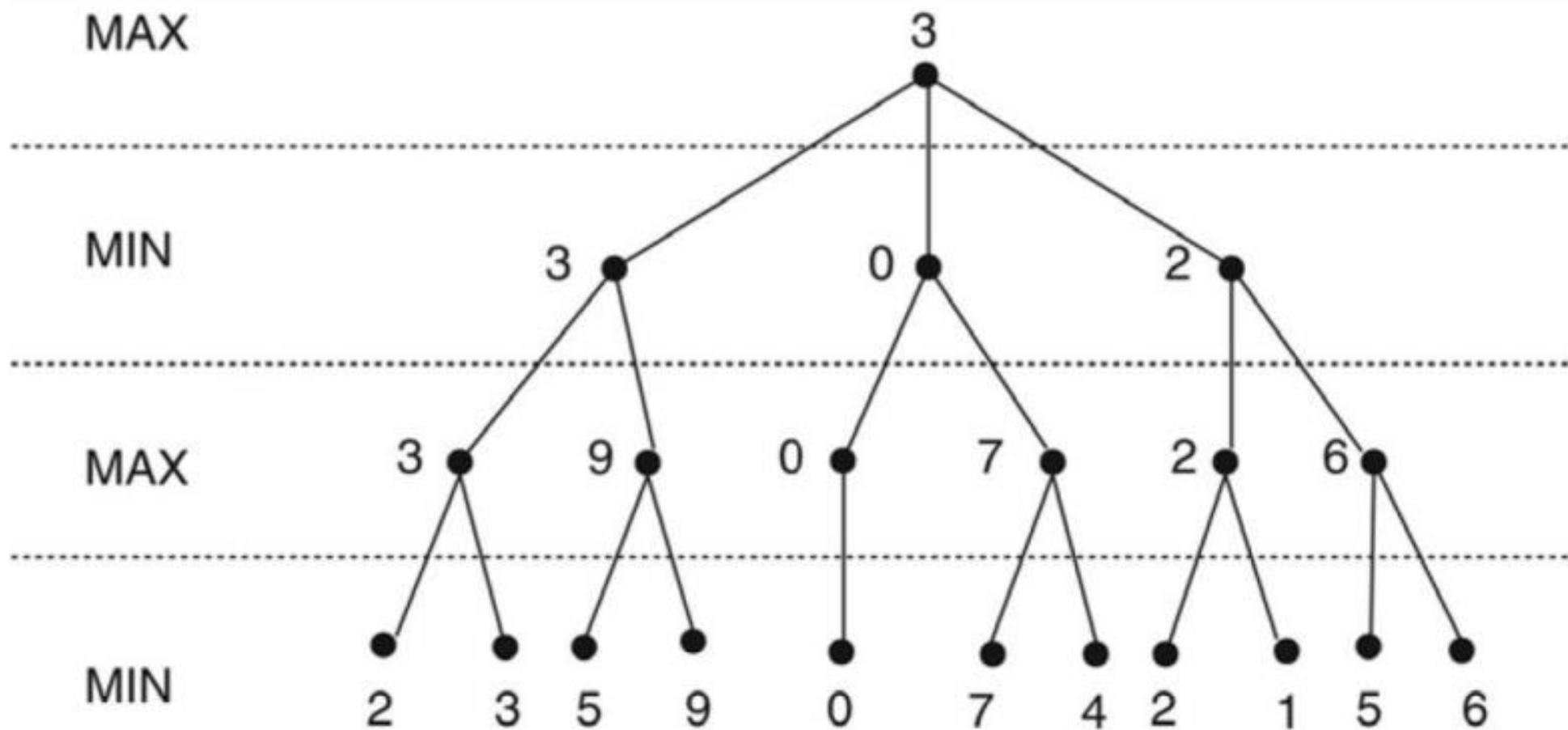
Điều kiện 2: Giá trị A < giá trị B

Điều kiện 3: X, Y, .., Z ở vị trí Min

Bỏ những cây con có gốc là X, Y, ..., Z



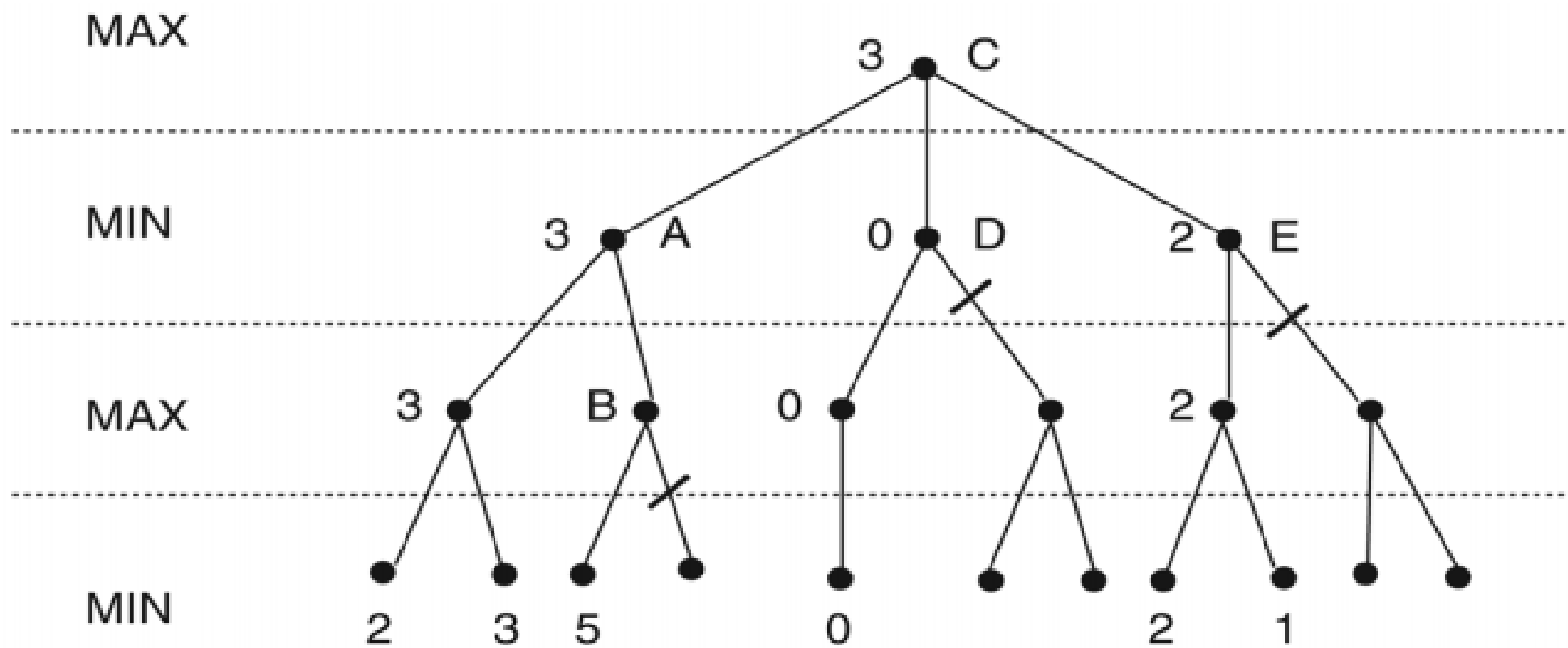
Thủ tục Minimax





Cắt tỉa Alpha - Beta

VD trong trường hợp sau:





Biểu diễn tri thức



GV: LÊ HOÀNG AN
KHOA: CÔNG NGHỆ THÔNG TIN



Biểu diễn tri thức

1. Tri thức và dữ liệu
2. Các loại tri thức
3. Biểu diễn tri thức



Tri thức và dữ liệu

- Dữ liệu: dạng thô tác động lên giác quan → không có nghĩa
- Dữ liệu được tổ chức → con người hiểu được → thông tin
- Công thức biểu diễn các thông tin → tri thức



Các loại tri thức

- Sự kiện: khẳng định về sự kiện, hiện tượng, khái niệm
- Mô tả: cho biết cảm nhận, cảm thấy, cấu tạo của đối tượng, sự kiện, ...
- Thủ tục: các quy trình, cách giải quyết vấn đề
- Heuristic: các ước lượng, phỏng đoán



Biểu diễn tri thức

- Logic mệnh đề
- Logic vị từ



Logic mệnh đề

- Mệnh đề: khẳng định hoặc đúng hoặc sai
- Phép tính: phủ định, hội, tuyển, kéo theo, tương đương
- Công thức biến đổi tương đương



Logic vị từ

Dạng biểu diễn vị từ

- $P(A)$: đối tượng A có thuộc tính P
- $P(A,B)$: đối tượng A có thuộc tính P đối với B
- Hằng: chữ thường
- Biến: chữ hoa
- Lượng từ phổ biến, tồn tại
- Biểu diễn câu bằng logic vị từ



Ví dụ

- “Bất cứ ai thi đậu nhờ trúng tử thì sung sướng”
 $\rightarrow \forall X, (ThiDau(X) \wedge TrungTu(X)) \Rightarrow SungSuong(X)$
- “John thông minh nhưng lười biếng”
 $\rightarrow ThongMinh(John) \wedge LuoiBieng(John)$
- “Ca sĩ thì có nhiều Fan và nhận nhiều Email”
 $\rightarrow \forall X, CaSi(X) \Rightarrow (NhieuFan(X) \wedge NhieuEmail(X))$



Hợp giải

Cho tập dữ liệu F , hãy chứng minh P

1. B1: Chuyển các câu trong F về dạng mệnh đề
2. B2: Lấy phủ định của P , bổ sung vào các mệnh đề có được ở bước 1
3. B3: Lặp lại cho đến khi tìm thấy mâu thuẫn hoặc không thể tiếp tục:
 - B3.1: Chọn cặp mệnh đề có chứa L và $\neg L$
 - B3.2: Hợp giải
 - B3.3: Kết quả rỗng \rightarrow mâu thuẫn, ngược lại bổ sung kết quả thu được và tập các mệnh đề



Các ví dụ



HỆ CHUYÊN GIA DỰA TRÊN LUẬT



GV: LÊ HOÀNG AN
KHOA: CÔNG NGHỆ THÔNG TIN



Mở đầu

- Giải quyết vấn đề dựa trên tri thức chuyên gia
- Thiết kế:
 - Thu thập tri thức: lý thuyết, kinh nghiệm, kỹ xảo, phương pháp làm tắt, chiến lược Heuristic đã tích lũy
 - Cài đặt vào hệ thống cho máy “hiểu” và có thể mô phỏng theo cách thức chuyên gia con người làm việc



Mở đầu

- Khả năng hệ chuyên gia:
 - Cung cấp sự kiểm tra đối với các quá trình suy luận của chúng, bằng cách hiển thị các bước trung gian và bằng cách trả lời câu hỏi về quá trình giải.
 - Cho phép sửa đổi dễ dàng, cả khi thêm và xóa các kỹ năng giải quyết vấn đề vào cơ sở tri thức (knowledge based).
 - Suy luận một cách heuristic, sử dụng tri thức (thường là không hoàn hảo) để tìm lời giải hữu ích cho vấn đề.

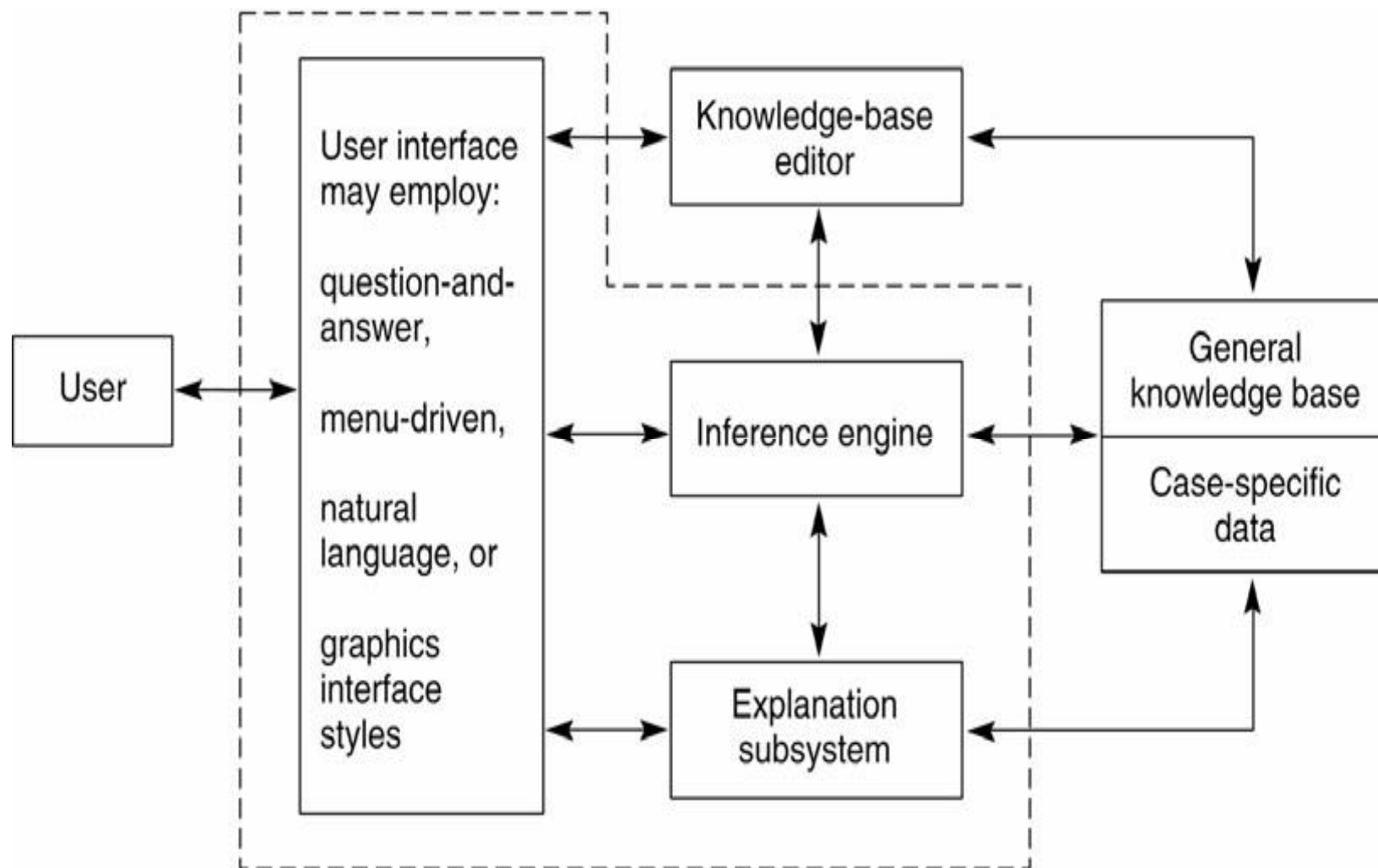


Mở đầu

- Các loại vấn đề đã giải quyết được:
 - Diễn giải
 - Dự đoán
 - Chẩn đoán
 - Thiết kế
 - Lập kế hoạch
 - Theo dõi
 - Bắt lỗi và sửa lỗi
 - Hướng dẫn
 - Điều khiển



Thiết kế HCG dựa trên luật





Thiết kế HCG dựa trên luật

1. Giao diện người dùng:

- Tương tác giữa người và HCG
- Yêu cầu: đơn giản hóa việc giao tiếp
- Kiểu giao diện: hỏi và trả lời, điều khiển bởi trình đơn, ngôn ngữ tự nhiên hay đồ họa, ...



Thiết kế HCG dựa trên luật

2. Cơ sở tri thức tổng quát

- Trái tim của HCG
- Chứa tri thức giải quyết vấn đề
- Biểu diễn dưới dạng các luật if... then....
- Bao gồm tri thức tổng quát (general knowledge) + tình huống cụ thể (case-specific).



Thiết kế HCG dựa trên luật

3. Động cơ suy diễn

- Áp dụng tri thức giải quyết bài toán
- Là trình thông dịch cho CS tri thức



Thiết kế HCG dựa trên luật

4. Hệ con giải thích

- Giải thích quá trình suy luận của HCG
- Biện minh cho kết luận của hệ thống (How?)
- Giải thích vì sao hệ thống cần dữ liệu đó (Why?)



Thiết kế HCG dựa trên luật

5. Trình soạn thảo CS tri thức

- Xác định, hiệu chỉnh lỗi
- Truy xuất thông tin cung cấp bởi hệ con giải thích
- Bổ sung tri thức mới
- ...



Các vấn đề phù hợp xây dựng HCG

- Sự cần thiết phải có một giải pháp biện minh cho chi phí và sức lực của việc xây dựng HCG
- Hiểu biết chuyên môn của con người không có sẵn ở mọi nơi
- Vấn đề có thể giải quyết được bằng cách sử dụng các kỹ thuật suy luận ký hiệu
- Phạm vi xác định vấn đề được cấu trúc tốt, không đòi hỏi sự suy luận theo lẽ thường



Các vấn đề phù hợp xây dựng HCG

- Vấn đề không thể giải quyết được bằng phương pháp truyền thống
- Có sự hợp tác và hiểu ý nhau giữa các chuyên gia
- Vấn đề cần giải quyết phải có kích thước và quy mô đúng mức



Ví dụ về HCG

- HCG chẩn đoán lỗi trên xe hơi
- Hệ chuyên gia R1/XCON



Ưu điểm của HCG

- Khả năng sử dụng trực tiếp các tri thức thực nghiệm của các chuyên gia.
- Tính module của luật làm cho việc xây dựng và bảo trì luật dễ dàng.
- Có thể thực hiện tốt trong các lĩnh vực hạn hẹp.
- Có tiện ích giải thích tốt.
- Các luật ánh xạ một cách tự nhiên vào không gian tìm kiếm trạng thái.
- Dễ dàng theo dõi một chuỗi các luật và sửa lỗi.
- Sự tách biệt giữa tri thức và điều khiển giúp đơn giản hóa quá trình phát triển Hệ chuyên gia



Khuyết điểm của HCG

- Các luật mang tính Heuristic rất cao
- Các luật Heuristic “dễ vỡ”
- Giải thích chứ không chứng minh
- Tri thức chỉ giới hạn gắn liền với công việc
- Khó bảo trì



MÁY HỌC



GV: LÊ HOÀNG AN
KHOA: CÔNG NGHỆ THÔNG TIN



Định nghĩa

- Học được định nghĩa như là bất cứ **sự thay đổi** nào trong một hệ thống cho phép nó **tiến hành tốt hơn trong lần thứ hai** khi lặp lại cùng một nhiệm vụ hoặc với một **nhiệm vụ khác** rút ra từ cùng một quần thể các nhiệm vụ đó
- CTH chỉ khảo sát một phần nhỏ trong toàn bộ các ví dụ có thể → khái quát hóa những ví dụ chưa từng gặp



Các tiếp cận học

- Tiếp cận ký hiệu
- Tiếp cận mạng neuron hay kết nối
- Tiếp cận nổi trội hay di truyền và tiến hóa



Tiếp cận ký hiệu – ID3

- Biểu diễn khái niệm ở dạng cây quyết định
- Học cây quyết định từ một tập các dữ liệu rèn luyện
- Input: tập các ví dụ
- Output: cây quyết định phân loại đúng các ví dụ trong tập dữ liệu rèn luyện và hy vọng phân loại đúng cho cả các ví dụ chưa gặp



Tiếp cận kết nối (mạng Neuron)

- Bắt chước cách học của các hệ thần kinh sinh vật
- Không sử dụng ký hiệu tường minh
- Các mẫu của bài toán được mã hóa thành vector số



Tiếp cận kết nối

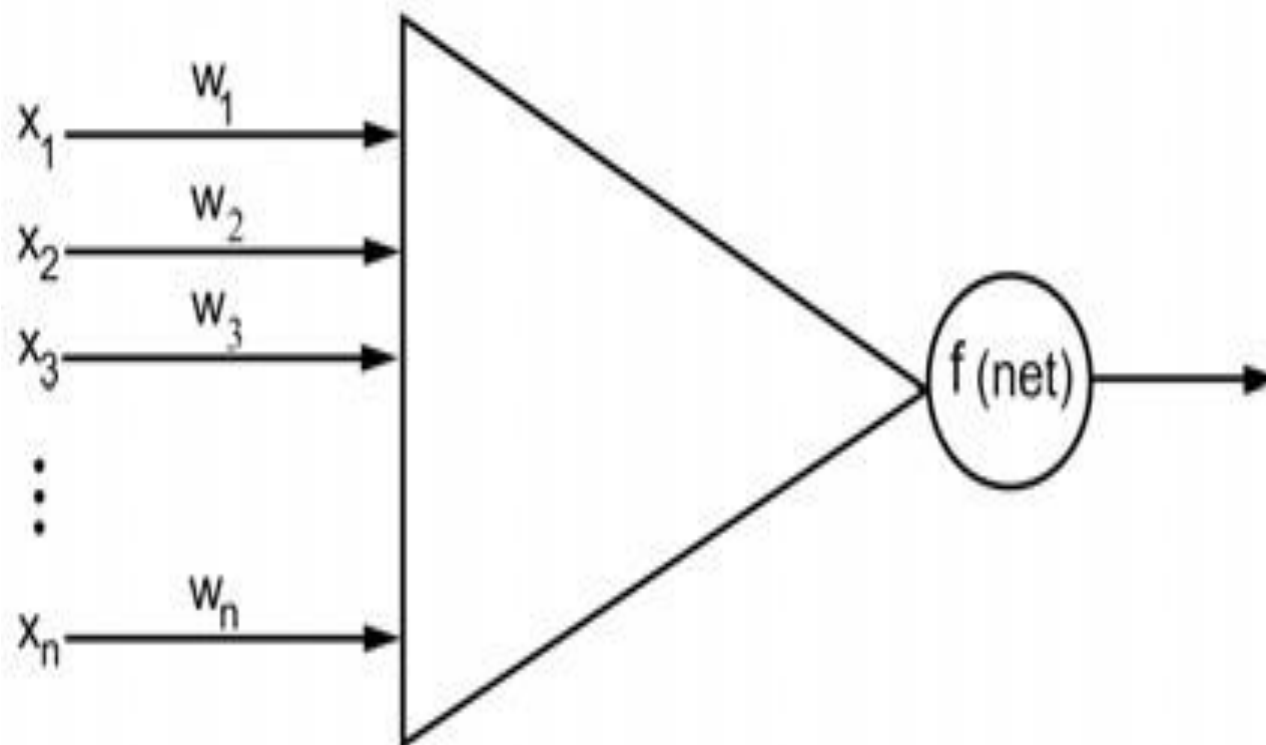
Các bài toán phù hợp

- Phân loại: quyết định giá trị đưa vào thuộc nhóm, loại nào
- Nhận dạng mẫu: nhận dạng các nhiễu trong dữ liệu
- Dự đoán, chẩn đoán: nhận dạng bệnh từ triệu chứng, ...
- Tối ưu: tìm tổ chức ràng buộc tốt nhất
- Lọc nhiễu: tìm ra các thành phần không quan trọng



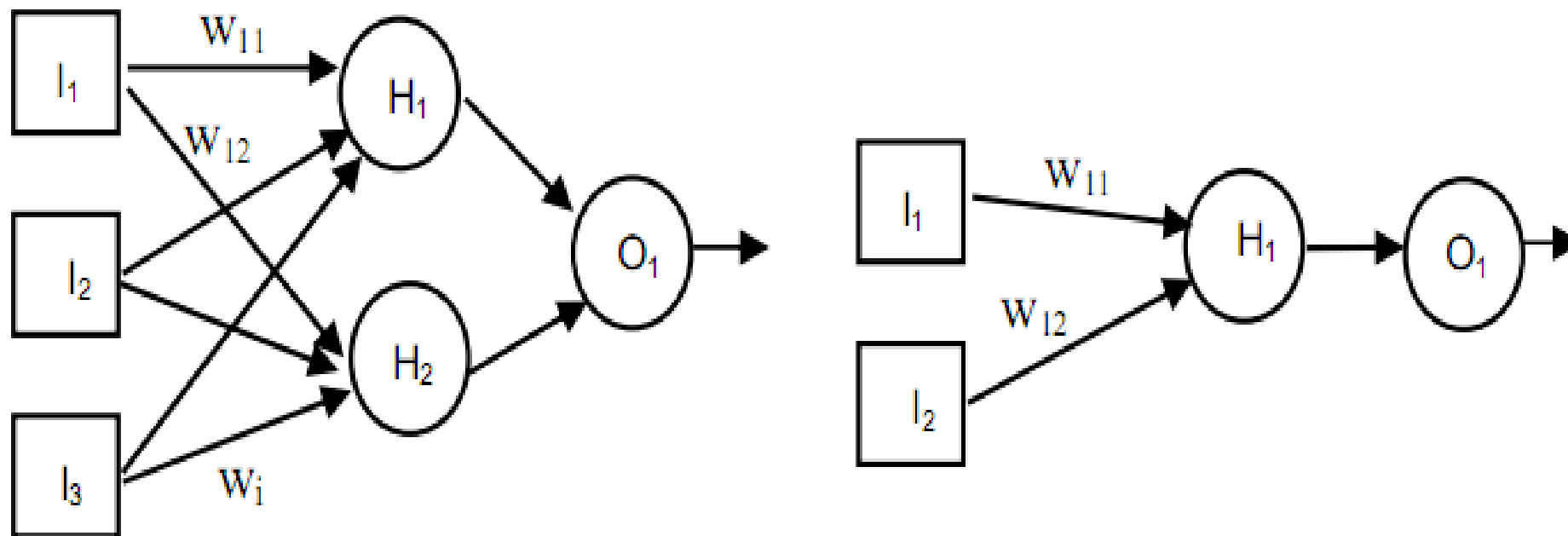
Neuron nhân tạo

- Các tín hiệu vào: các số rời rạc
- Các trọng số: mô tả sức mạnh kết nối
- Mức kích hoạt: tổng các giá trị đầu vào
- Hàm ngưỡng





Các đặc trưng của mạng Neuron





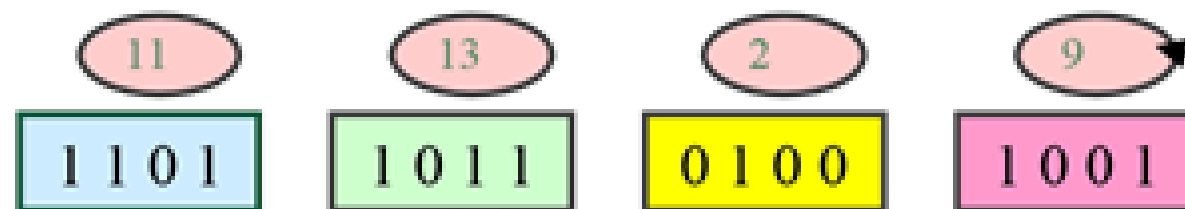
Tiếp cận xã hội và nổi trội

- Sự cạnh tranh trong một quần thể gồm các lời giải ứng viên đang tiến hóa của bài toán
- Hàm “thích nghi” đánh giá khả năng đóng góp cho lời giải tiếp theo



Toán tử di truyền

Thế hệ n



Độ thích nghi
(fitness)

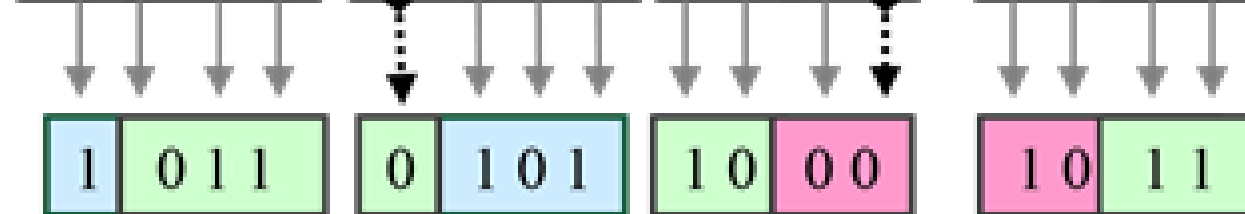
Tái sản xuất



Lai ghép



Đột biến



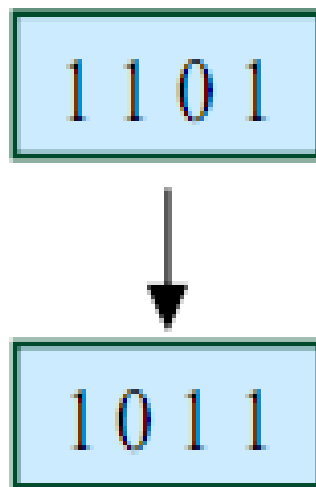
Thế hệ n+1



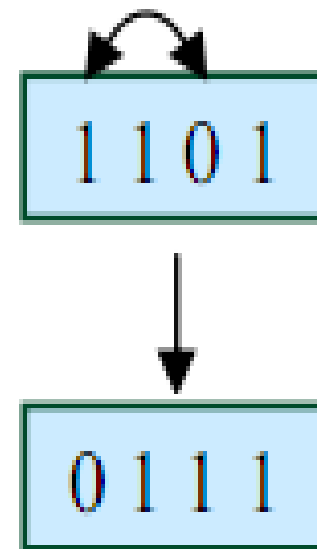


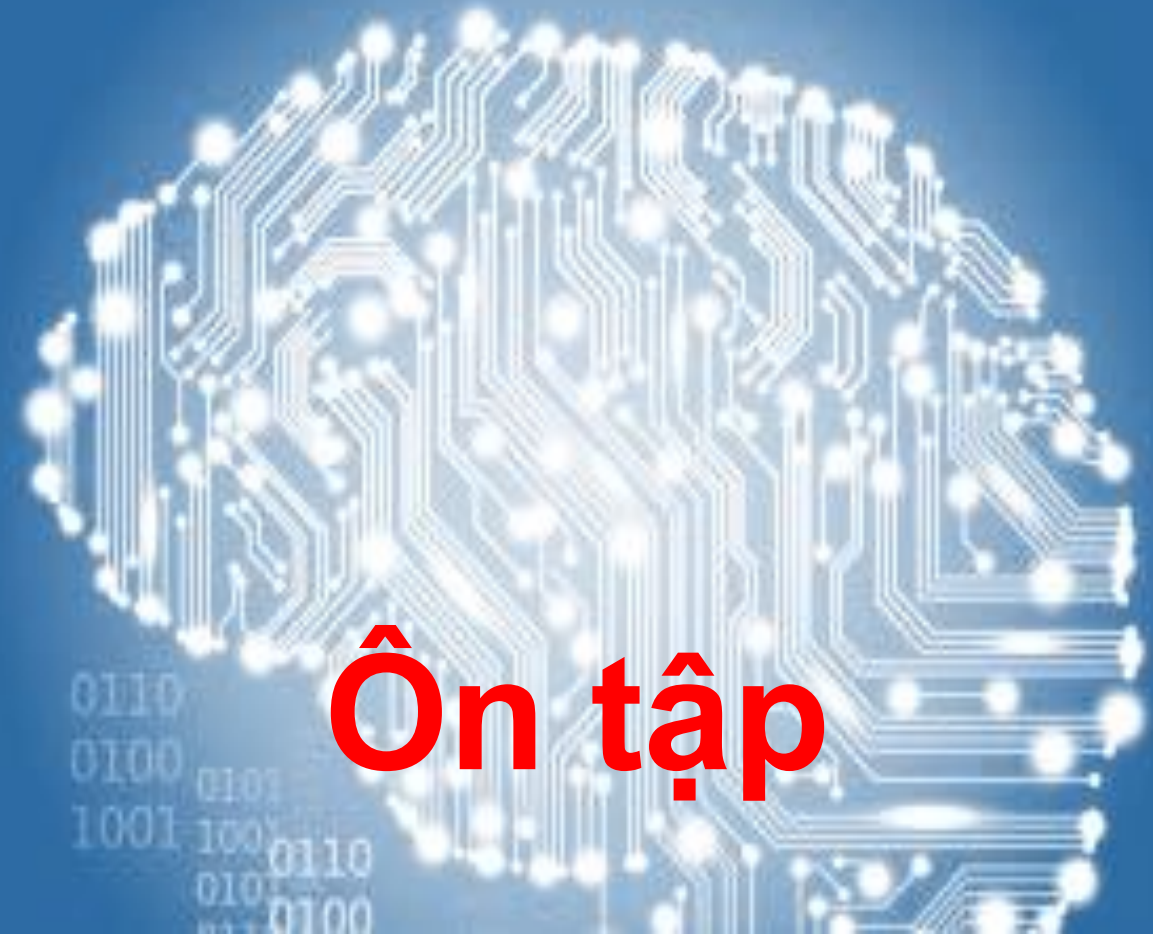
Toán tử di truyền

Đảo ngược



Trao đổi





Ôn tập



GV: LÊ HOÀNG AN
KHOA: CÔNG NGHỆ THÔNG TIN



Mục tiêu của ngành trí tuệ nhân tạo là gì?

- A. Nghiên cứu ý tưởng giúp máy tính có khả năng nhận thức, suy luận và phản ứng.
- B. Xây dựng máy tính có năng lực tư duy như con người.
- C. Xây dựng các máy tính thông minh.
- D. Tất cả các ý trên.



Ngôn ngữ lập trình đầu tiên dùng cho Trí tuệ nhân tạo là gì?

- A. COBOL
- B. FORTRAN
- C. LISP
- D. PROLOG



Phát biểu nào **không** là ứng dụng của trí tuệ nhân tạo?

- A. Phát triển hệ thống siêu máy tính.
- B. Phát triển các hệ chuyên gia.
- C. Mô hình hóa ngôn ngữ tự nhiên.
- D. Suy luận và chứng minh định lý tự động.



Việc một hệ chuyên gia **không thể nhớ lời giải của bài toán đã gặp trước đó** là vấn đề cần giải quyết trong lĩnh vực nào của trí tuệ nhân tạo?

- A. Mô hình hóa hoạt động con người
- B. Máy học
- C. Lập kế hoạch và robotics
- D. Hệ chuyên gia



Phương pháp tìm kiếm hướng từ dữ liệu nên áp dụng trong trường hợp nào sau đây?

- A. Mục tiêu đã được cho trong phát biểu bài toán.
- B. Mục tiêu dễ dàng được tìm thấy.
- C. Có một số lượng lớn các kết luận được tạo ra sau mỗi bước.
- D. Khó hình thành nên một đích hay một kết luận.



Điểm giống nhau giữa tìm kiếm leo núi và tìm kiếm sâu là gì?

- A. Luôn tìm ra lời giải (nếu có).
- B. Sử dụng Heuristic.
- C. Điều “đi thẳng”.
- D. Có quay lui.



Ưu điểm của tìm kiếm sâu so với tìm kiếm leo núi là gì?

A. Tìm ra lời giải gần gốc nhất

B. Có quay lui

C. “Đi thẳng” vào cây tìm kiếm

D. Sử dụng Heuristic



Trong không gian bài toán 4 thành phần $[N, A, S, GD]$, phát biểu nào sau đây là đúng?

- A. N là tập trạng thái bắt đầu.
- B. A là tập tất cả trạng thái.
- C. S là tập các cung.
- D. S là tập con của N .



Tìm kiếm theo chiều rộng có đặc điểm:

A. Tìm kiếm theo mức.

B. Tìm kiếm theo hướng được cho là có lời giải.

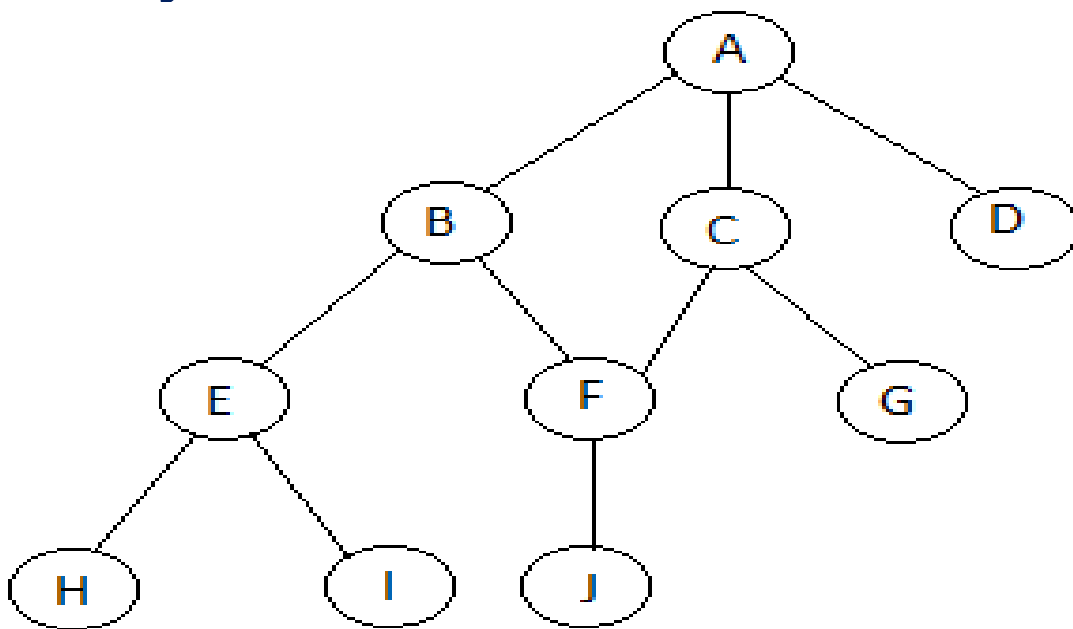
C. Đi sâu dần trong cây tìm kiếm, khi đến mức giới hạn thì quay lui.

D. Đi sâu dần trong cây tìm kiếm, quay lui khi gặp nút lá.



Áp dụng phương pháp tìm kiếm theo chiều sâu tìm đường đi đến G sẽ xét qua mấy nút?

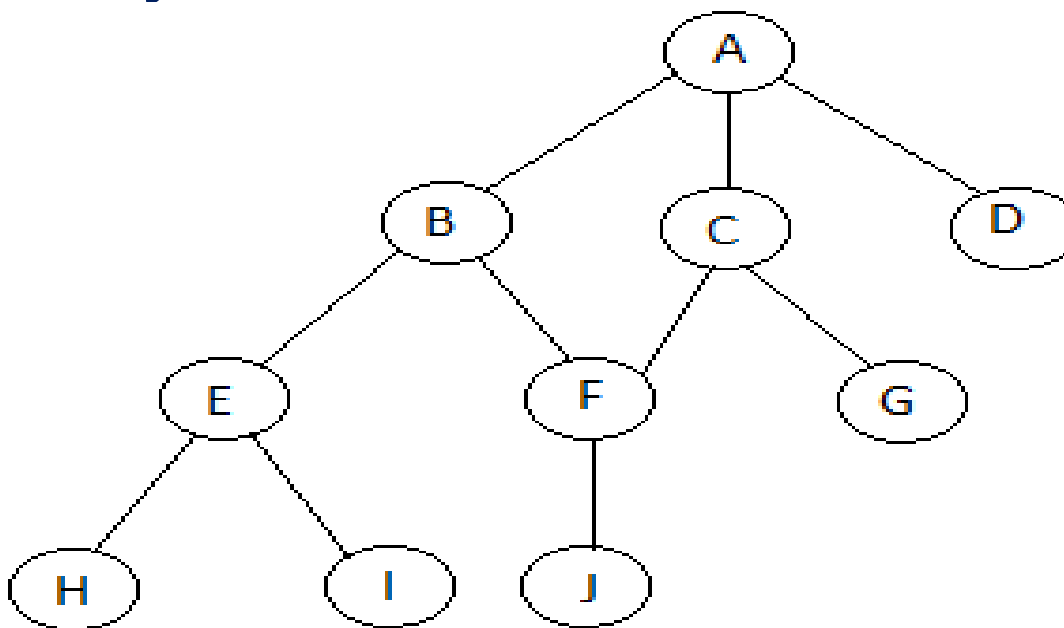
- A. 9
- B. 7
- C. 10
- D. 3





Áp dụng phương pháp tìm kiếm theo chiều sâu tìm đường đi đến F sẽ xét qua mấy nút?

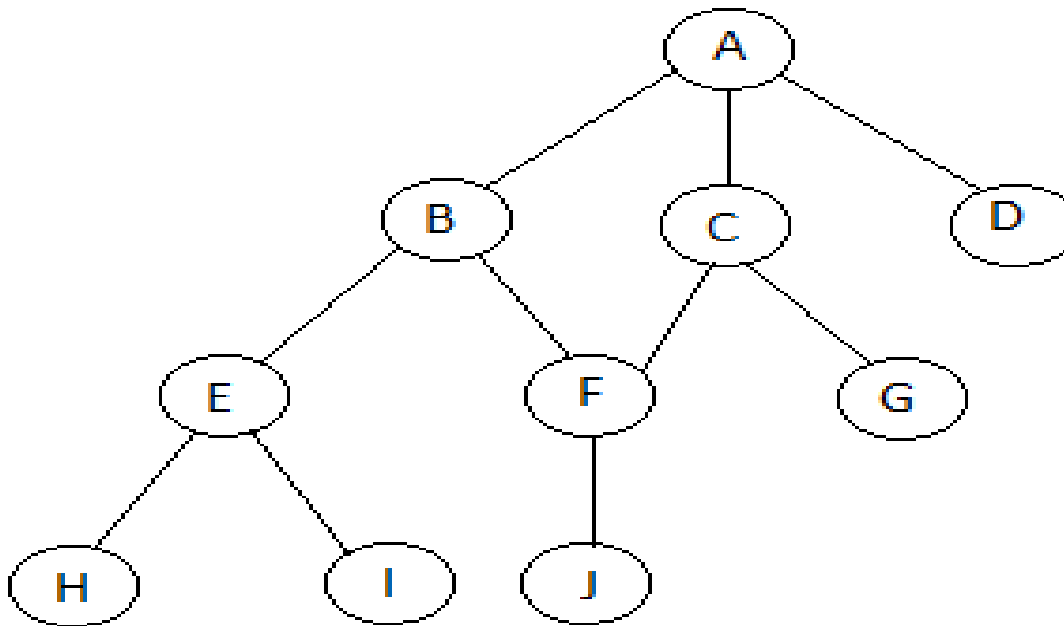
- A. 4
- B. 6
- C. 7
- D. 5





Áp dụng phương pháp tìm kiếm theo chiều rộng tìm đường đi đến G, sau khi xét B, nút tiếp theo được xét là

- A. A
- B. C
- C. F
- D. D





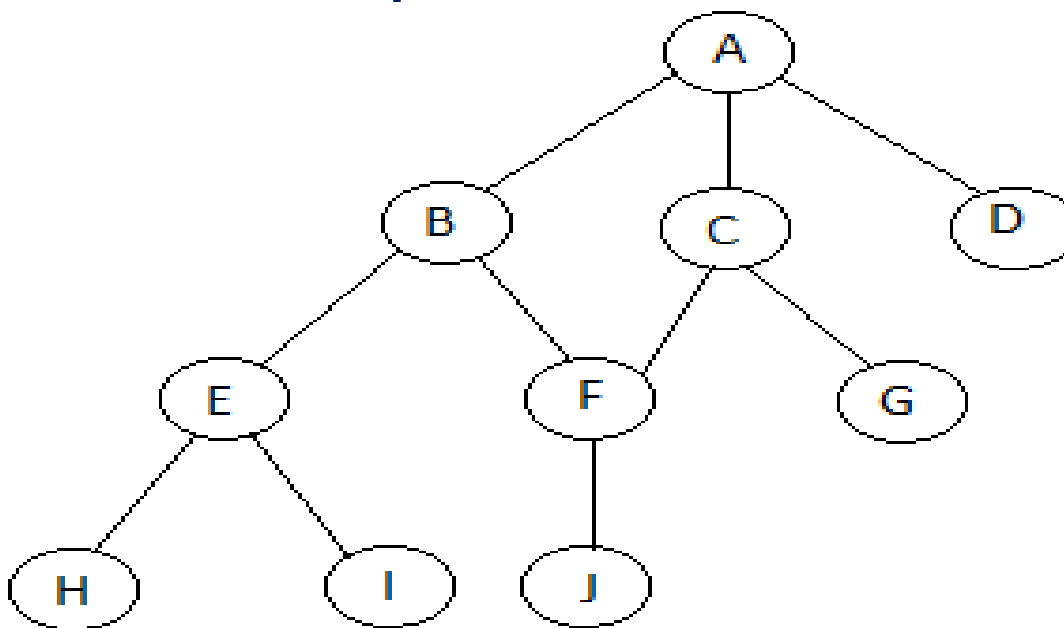
Áp dụng phương pháp tìm kiếm sâu dần với $k=3$ tìm đường đi đến J, sau khi xét E, nút tiếp theo được xét là

A. B

B. I

C. H

D. F





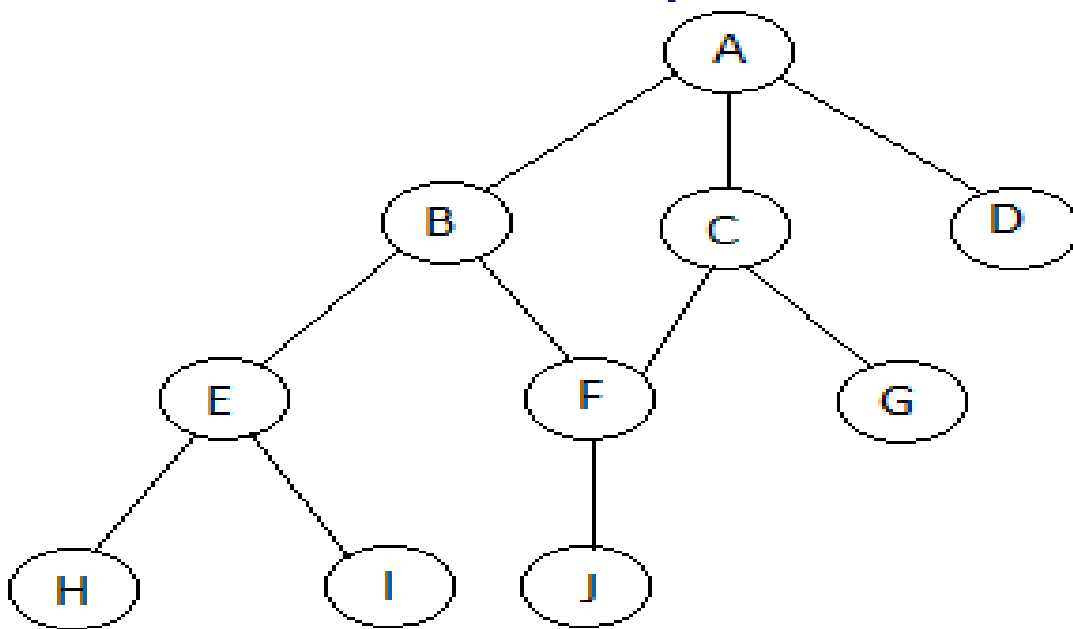
Áp dụng phương pháp tìm kiếm theo chiều rộng tìm đường đi đến G, sau khi xét B, danh sách Open là

A. [C, D, E, F]

B. [B, C, D]

C. [E, F, C, D]

D. [A, B]





Đặc điểm của tìm kiếm leo núi đơn giản là gì?

- A. Chắc chắn tìm ra được lời giải.
- B. Trạng thái được chọn là trạng thái tốt nhất được sinh ra.
- C. Chỉ “đi thẳng”, không quay lui.
- D. Chỉ là tìm kiếm mù, không sử dụng Heuristic.



Tìm kiếm ưu tiên tối ưu, danh sách Open có cấu trúc gì?

A. LIFO

B. FIFO

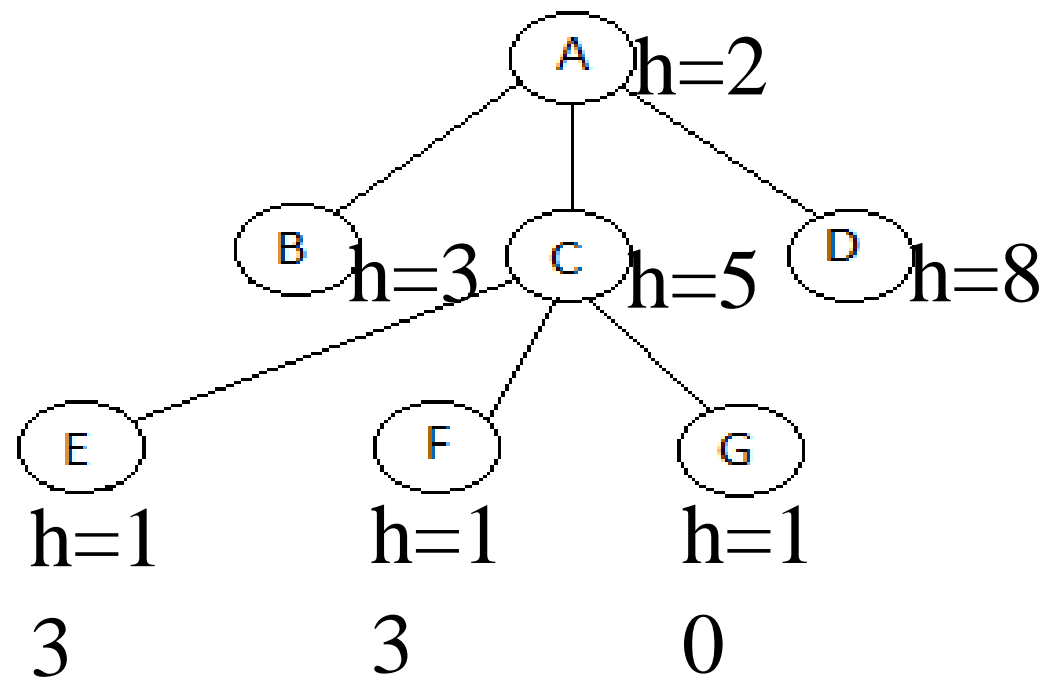
C. Danh sách tùy ý.

D. Hàng đợi ưu tiên.



Khi áp dụng phương pháp tìm kiếm leo núi đơn giản vào không gian trên với h phản ánh sự cản trở tiếp cận lời giải, sau khi xét A, bước tiếp theo thực hiện

- A. Xét D
- B. Xét C
- C. Xét B
- D. Dừng quá trình.





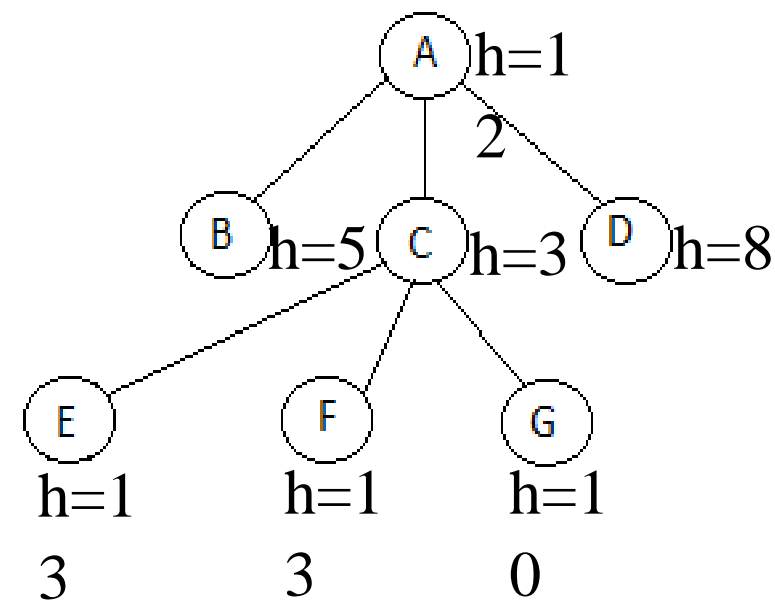
Khi áp dụng phương pháp tìm kiếm leo núi đơn giản vào không gian trên với h phản ánh sự cản trở tiếp cận lời giải để tìm đường đi từ A đến G, thứ tự các nút được xét là:

A. A – B – C – E – F – G

B. A – C – B – D – G

C. A – C – G

D. A – B, dừng quá trình





Biểu thức vị từ của sự kiện “Bất cứ ai thi đậu nhờ trúng tủ thì sung sướng” là biểu thức nào?

- A. $\forall X, ThiDau(X) \wedge TrungTu(X) \vee SungSuong(X)$
- B. $\forall X, ThiDau(X) \wedge TrungTu(X) \wedge SungSuong(X)$
- C. $\forall X, (ThiDau(X) \vee TrungTu(X)) \Rightarrow SungSuong(X)$
- D. $\forall X, (ThiDau(X) \wedge TrungTu(X)) \Rightarrow SungSuong(X)$



Khi sử dụng hợp giải để chứng minh: “Nam thông minh và giàu có” tại Bước 2, mệnh đề được bổ sung vào là:

- A. $\neg \text{ThôngMinh}(\text{Nam}) \wedge \text{Giàu}(\text{Nam})$
- B. $\neg \text{ThôngMinh}(\text{Nam}) \wedge \neg \text{Giàu}(\text{Nam})$
- C. $\text{ThôngMinh}(\text{Nam}) \wedge \text{Giàu}(\text{Nam})$
- D. $\neg \text{ThôngMinh}(\text{Nam}) \vee \neg \text{Giàu}(\text{Nam})$



Cấu trúc hệ chuyên gia dựa trên luật cơ bản bao gồm:

A. Cơ sở tri thức, Giao diện người dùng, Bộ máy suy diễn.

B. Cơ sở tri thức, Hệ con giải thích, Động cơ suy luận, Giao diện người dùng.

C. Cơ sở tri thức, Trình soạn thảo cơ sở tri thức, Bộ máy suy diễn, Giao diện người dùng.

D. Cơ sở tri thức, Giao diện người dùng, Bộ máy suy diễn, Động cơ suy luận.



- Phát biểu nào sai khi nói về ưu điểm của hệ chuyên gia?
- A. Xử lý tốt các trường hợp ngoài dự kiến.
 - B. Dễ dàng theo dõi một chuỗi các luật và sửa đổi.
 - C. Thực hiện tốt trong một lĩnh vực hẹp.
 - D. Khả năng sử dụng trực tiếp các tri thức thực nghiệm của chuyên gia.



Hệ chuyên gia nào sau đây được thiết kế để phỏng đoán cấu trúc của các phân tử hữu cơ?

- A. XCON
- B. PROSPECTOR
- C. DENDRAL
- D. MYCIN



Hệ chuyên gia không giải quyết được loại vấn đề nào sau đây?

- A. Diễn giải
- B. Dự đoán
- C. Chẩn đoán
- D. Mô phỏng



Phát biểu nào sai khi nói về tiếp cận ký hiệu?

- A. Trong tiếp cận ký hiệu, vấn đề được biểu diễn dưới dạng ký hiệu.
- B. Hàm ngưỡng là thành phần cơ bản trong tiếp cận ký hiệu.
- C. Tiếp cận ký hiệu không là dạng mô phỏng sự tiến hóa trong tự nhiên.
- D. Giải thuật ID3 được sử dụng rộng rãi trong tiếp cận ký hiệu.



Trên lời giải ứng viên, toán tử di truyền đảo ngược thực hiện

- A. Đảo ngược thứ tự các bit.
- B. Trao đổi các thành phần của 2 lời giải.
- C. Hoán đổi vị trí 2 bit đầu cuối.
- D. Đảo các bit 0 thành 1 và ngược lại.



Một Neuron nhân tạo bao gồm các thành phần nào?

A. Các tín hiệu đầu vào

B. Tập các trọng số

C. Mức kích hoạt

D. Tất cả các ý trên



Phát biểu nào không là đặc trưng của mạng Neuron?

- A. Hình thái mạng
- B. Độ lớn của quần thể
- C. Sơ đồ mã hóa
- D. Giải thuật học



Dạng chuẩn hội của phát biểu “Nếu P và Q thì R” là biểu thức nào sau đây?

A. $P \vee Q \Rightarrow R$

B. $(P \vee \neg R) \wedge (Q \vee \neg R)$

C. $(P \wedge \neg R) \vee (Q \vee \neg R)$

D. $\neg P \vee \neg Q \vee R$



Phát biểu “Máy tính bao gồm phần cứng và phần mềm”
là loại tri thức gì?

- A. Tri thức thủ tục.
- B. Tri thức sự kiện.
- C. Tri thức Heuristic.
- D. Tri thức mô tả.