

ĐỀ THI GIỮA KỲ

Nhập môn Trí tuệ nhân tạo – Lớp MAT3508 3 và MAT3508 4

Học kỳ I – Năm học 2026–2027 – Ngày thi: 28/10/2026

Thời gian: 90 phút (không kể thời gian phát đề)

Câu 1 (1 điểm)

- (a) (0,5 điểm) Một tác tử có thể được mô tả một cách tổng quát như thế nào?

How can an agent be generally described?

- (b) (0,5 điểm) Vì sao một tác tử tất định có bộ nhớ không phải là một hàm (theo nghĩa toán học) từ tập các đầu vào đến tập các đầu ra?

Why is a deterministic agent with memory not a function (in the mathematical sense) from the set of inputs to the set of outputs?

Câu 2 (1,5 điểm)

- (a) (0,5 điểm) Dạng chuẩn hội (CNF) là gì?

What is conjunctive normal form (CNF)?

- (b) (1 điểm) Biến đổi công thức sau về dạng chuẩn hội:

Transform the following formula into conjunctive normal form:

$$(A \wedge B) \Leftrightarrow (A \vee B)$$

Câu 3 (1,5 điểm)

- (a) (0,5 điểm) Phép hợp nhất tổng quát nhất (most general unifier – MGU) là gì?

What is a most general unifier (MGU)?

- (b) (1 điểm) Hợp nhất (nếu có thể) hai biểu thức sau, trong đó x, y là biến. Trình bày từng bước thuật toán hợp nhất, cho biết MGU và các biểu thức thu được sau khi hợp nhất.

Unify (if possible) the following expressions, where x and y are variables. Show each step of the unification algorithm, and give the MGU and the resulting expressions.

$$p(x, f(x)), \quad p(y, y)$$

Câu 4 (2 điểm)

Một nông dân cần đưa một con sói, một con dê và một cây bắp cải từ bờ trái sang bờ phải của một con sông. Thuyền chỉ chở được nông dân cùng nhiều nhất một trong ba thứ trên. Khi không có nông dân ở cùng bờ, sói sẽ ăn dê và dê sẽ ăn bắp cải. Mỗi trạng thái được biểu diễn bằng hạng thức $state(Farmer, Wolf, Goat, Cabbage)$, các thành phần nhận giá trị left (bờ trái) hoặc right (bờ phải). Cho chương trình GNU Prolog sau:

A farmer needs to take a wolf, a goat and a cabbage from the left bank to the right bank of a river. The boat can carry only the farmer and at most one of the three. When the farmer is not on the same bank, the wolf eats the goat and the goat eats the cabbage. Each state is represented by the term $state(Farmer, Wolf, Goat, Cabbage)$, whose arguments are left or right (the bank). Consider the following GNU Prolog program:

```

start :- action(state(left,left,left,left),
                state(right,right,right,right)).

action(Start,Goal) :-
    plan(Start,Goal,[Start],Path),
    nl, write('Solution:'), nl,
    write(Path), nl.

plan(Start,Goal,Visited,Path) :-
    go(Start,Next),
    safe(Next),
    \+ member(Next,Visited),      % not(member(...))
    plan(Next,Goal,[Next|Visited],Path).
plan(Goal,Goal,Path,Path).

go(state(X,X,Z,K),state(Y,Y,Z,K)) :- across(X,Y). % farmer, wolf
go(state(X,W,X,K),state(Y,W,Y,K)) :- across(X,Y). % farmer, goat
go(state(X,W,Z,X),state(Y,W,Z,Y)) :- across(X,Y). % farmer, cabbage
go(state(X,W,Z,K),state(Y,W,Z,K)) :- across(X,Y). % farmer

across(left,right).
across(right,left).

safe(state(_,W,Z,K)) :- across(W,Z), across(Z,K).
safe(state(B,B,B,_)).
safe(state(B,_,B,B)).

```

(a) (0,5 điểm) Cần nhập truy vấn nào để chạy chương trình?

What query should be entered to run the program?

(b) (0,5 điểm) Chạy chương trình trên, ta được kết quả:

Running the program above gives the following output:

```

Solution:
[state(right,right,right,right),state(left,right,left,right),
 state(right,right,left,right),state(left,right,left,left),
 state(right,right,right,left),state(left,left,right,left),
 state(right,left,right,left),state(left,left,left,left)]

```

Nhận xét về thứ tự các trạng thái trong danh sách. Từ đó, trình bày lời giải thành dãy các lần qua sông **theo đúng thứ tự thực hiện**, mỗi lần qua sông viết thành một câu, chẳng hạn:

Farmer and wolf cross from left to right.

Comment on the order of the states in the list. Based on that, write the solution as a sequence of river crossings in the order they are carried out, one sentence per crossing, for example:

Farmer and wolf cross from left to right.

(c) (1 điểm) Để chương trình tự in lời giải thành các câu như ở ý (b) thay vì danh sách trạng thái, ta thay dòng `write(Path), nl.` trong vị từ `action` bằng `write_path(Path).` và thêm vị từ `write_path` vào cuối chương trình như sau:

To make the program print the solution as sentences like those in (b) instead of a list of states, we replace the line `write(Path), nl.` in the predicate `action` by `write_path(Path).` and add the predicate `write_path` at the end of the program as follows:

```

action(Start,Goal) :-
    plan(Start,Goal,[Start],Path),
    nl, write('Solution:'), nl,
    write_path(Path).                % replaces write(Path), nl

% ... plan, go, across, safe: unchanged ...

write_path([_]).
write_path([S2,S1|Rest]) :-
    write_path([S1|Rest]),
    write_move(S1,S2).

% ===== BEGIN: Complete the clauses of write_move/2 below =====
% (1) Farmer crosses from <X> to <Y>
% (2) Farmer and wolf cross from <X> to <Y>
% (3) Farmer and goat cross from <X> to <Y>
% (4) Farmer and cabbage cross from <X> to <Y>
% ===== END: write_move/2 =====

```

Hãy hoàn thiện chương trình: tại mỗi vị trí có chú thích (1)–(4), viết một mệnh đề của vị từ `write_move(State1,State2)`, trong đó `State1`, `State2` là trạng thái ngay trước và ngay sau một lần qua sông. Mỗi mệnh đề in đúng một dòng theo mẫu ở chú thích (<X>: bờ xuất phát, <Y>: bờ đến) và chỉ dùng `write/1`, `nl/0`. Bài làm chỉ cần ghi các mệnh đề này.

Complete the program: at each position marked by a comment (1)–(4), write one clause of the predicate `write_move(State1,State2)`, where `State1` and `State2` are the states just before and just after a river crossing. Each clause prints exactly one line following the pattern in its comment (<X>: bank of departure, <Y>: bank of arrival) and uses only `write/1` and `nl/0`. In your answer, write only these clauses.

Câu 5 (2 điểm)

Xét bài toán 8-puzzle với trạng thái đầu và trạng thái đích dưới đây (ô màu xám là ô trống; mỗi nước đi di chuyển ô trống sang một ô kề: lên (U), xuống (D), trái (L) hoặc phải (R)).

Consider the 8-puzzle with the start and goal states below (the grey square is the blank; a move shifts the blank to an adjacent square: up (U), down (D), left (L) or right (R)).

Trạng thái đầu

1		3
4	2	6
7	5	8

Trạng thái đích

1	2	3
4	5	6
7	8	

Dùng tìm kiếm theo chiều rộng (breadth-first search), hãy vẽ cây tìm kiếm từ trạng thái đầu cho đến khi tìm thấy trạng thái đích và cho biết đường đi tìm được (dãy nước đi của ô trống). Sinh các nút con theo thứ tự U, D, L, R, tránh chu trình độ dài 2 (không sinh nước đi quay lại trạng thái cha), và đánh số các nút theo thứ tự được sinh ra. Kiểm tra đích ngay khi sinh ra mỗi nút.

Using breadth-first search, draw the search tree from the start state until the goal state is found, and give the path found (the sequence of moves of the blank). Generate successors in the order U, D, L, R, avoiding cycles of length 2, and number the nodes in the order in which they are generated. Apply the goal test to each node as soon as it is generated.

Câu 6 (2 điểm)

Cần dự đoán thời tiết buổi chiều dựa vào một vài quan sát thời tiết đơn giản vào buổi sáng cùng ngày. Mô hình xác suất đầy đủ, tức phân phối xác suất đồng thời của ba biến *Sky*, *Bar* và *Prec*, được cho trong bảng dưới đây.

You are to predict the afternoon weather from a few simple weather observations made in the morning of the same day. The complete probabilistic model, i.e. the joint probability distribution of the three variables Sky, Bar and Prec, is given in the table below.

<i>Sky</i>	<i>Bar</i>	<i>Prec</i>	$P(\text{Sky}, \text{Bar}, \text{Prec})$
Clear	Rising	Dry	0,40
Clear	Rising	Raining	0,07
Clear	Falling	Dry	0,08
Clear	Falling	Raining	0,10
Cloudy	Rising	Dry	0,09
Cloudy	Rising	Raining	0,11
Cloudy	Falling	Dry	0,03
Cloudy	Falling	Raining	0,12

Sky: Bầu trời buổi sáng quang đãng (clear) hay nhiều mây (cloudy).

The sky is clear or cloudy in the morning.

Bar: Khí áp kế buổi sáng báo áp suất tăng (rising) hay giảm (falling).

The barometer is rising or falling in the morning.

Prec: Buổi chiều trời mưa (raining) hay khô ráo (dry).

It is raining or dry in the afternoon.

- (a) (0,5 điểm) Phát biểu công thức xác suất có điều kiện $P(A | B)$, với $P(B) > 0$.

State the formula for the conditional probability $P(A | B)$, where $P(B) > 0$.

- (b) (0,5 điểm) Tính $P(\text{Prec} = \text{dry} | \text{Sky} = \text{clear}, \text{Bar} = \text{rising})$.

Compute $P(\text{Prec} = \text{dry} | \text{Sky} = \text{clear}, \text{Bar} = \text{rising})$.

- (c) (1 điểm) Mỗi buổi sáng, một thiết bị cảm biến dự báo buổi chiều có mưa hay không. Gọi S là sự kiện thiết bị báo mưa. Khi buổi chiều trời mưa, thiết bị báo mưa với xác suất $P(S | \text{Prec} = \text{raining}) = 0,9$; khi buổi chiều khô ráo, thiết bị vẫn báo mưa với xác suất $P(S | \text{Prec} = \text{dry}) = 0,2$. Dùng công thức Bayes, tính xác suất buổi chiều trời mưa khi thiết bị báo mưa, $P(\text{Prec} = \text{raining} | S)$.

Every morning, a sensor predicts whether it will rain in the afternoon. Let S be the event that the sensor predicts rain. When it rains in the afternoon, the sensor predicts rain with probability $P(S | \text{Prec} = \text{raining}) = 0.9$; when the afternoon is dry, it still predicts rain with probability $P(S | \text{Prec} = \text{dry}) = 0.2$. Using Bayes' formula, compute the probability that it rains in the afternoon given that the sensor predicts rain, $P(\text{Prec} = \text{raining} | S)$.

— HẾT —