

**ỦY BAN NHÂN DÂN TP HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC SÀI GÒN**

**KHOA CÔNG NGHỆ THÔNG TIN**



# **SEMINAR CHUYÊN ĐỀ**

**CHUYÊN ĐỀ: LẬP TRÌNH LOGIC VÀ ỨNG DỤNG**

## **BÀI TẬP BỔ SUNG**

**Lớp môn học: DKP1201**

**Họ tên sinh viên: Lý Kiến Huy**

**Mã số sinh viên: 3120410207**

**Giảng viên hướng dẫn: PGS.TS Nguyễn Tuấn Đăng**

.....

**TP. HCM tháng 11 /2024**

**Đề bài:** cho 3 bình nước ((bình 1: 8 lít, bình 2: 5 lít, bình 3: 3 lít), yêu cầu đong 4 lít).

Bài làm

**-Số lượng đối tượng tham gia vào trạng thái của bài toán:**

+Dung tích tối đa của bình 1: 8 lít

+Dung tích tối đa của bình 2: 5 lít

+Dung tích tối đa của bình 3: 3 lít

**-Biểu diễn trạng thái của bài toán:**

+Định nghĩa một trạng thái: state(bình 1,bình 2,bình 3)

Mỗi trạng thái được biểu diễn bởi lượng nước hiện có trong bình 1, bình 2,bình 3

Trạng thái ban đầu: state(0,0,0) (cả 3 bình đều rỗng)

Trạng thái đích: state(4,x,x), trong đó x là lượng nước bất kỳ trong bình 2 và bình 3

**-Các thao tác/ hành động có thể tác động lên các trạng thái:**

1. Đổ đầy bình 1
2. Đổ đầy bình 2
3. Đổ đầy bình 3
4. Làm rỗng bình 1
5. Làm rỗng bình 2
6. Làm rỗng bình 3
7. Chuyển nước từ bình 1 sang bình 2 cho đến khi bình 1 rỗng hoặc bình 2 đầy
8. Chuyển nước từ bình 1 sang bình 3 cho đến khi bình 1 rỗng hoặc bình 3 đầy
9. Chuyển nước từ bình 2 sang bình 3 cho đến khi bình 2 rỗng hoặc bình 3 đầy
10. Chuyển nước từ bình 3 sang bình 2 cho đến khi bình 3 rỗng hoặc bình 2 đầy
11. Chuyển nước từ bình 3 sang bình 1 cho đến khi bình 3 rỗng hoặc bình 1 đầy
12. Chuyển nước từ bình 2 sang bình 1 cho đến khi bình 2 rỗng hoặc bình 1 đầy

**-Mô tả không gian trạng thái:**

+Không gian trạng thái bao gồm tất cả các cấu hình có thể của lượng nước trong ba bình,từ state(0,0,0) đến state(4,5,3).

+Chỉ các trạng thái hợp lệ (lượng nước không vượt quá dung tích của bình tương ứng) được xem xét trong không gian trạng thái

-Áp dụng thuật toán DFS để tìm nghiệm:

Khởi tạo:

OPEN:[state(0,0,0)]

CLOSED:[]

Bước 1:

Trạng thái đang xét : state(0, 0, 0)

Trạng thái sinh ra: state(0, 0, 3), state(8, 0, 0), state(0, 5, 0)

OPEN: [state(0, 0, 3), state(8, 0, 0), state(0, 5, 0)]

CLOSED: [state(0, 0, 0)]

Bước 2:

Trạng thái đang xét : state(0, 0, 3)

Trạng thái sinh ra: state(0, 3, 0), state(3, 0, 0), state(0, 0, 0), state(8, 0, 3), state(0, 5, 3)

OPEN: [state(0, 3, 0), state(3, 0, 0), state(8, 0, 3), state(0, 5, 3), state(8, 0, 0), state(0, 5, 0)]

CLOSED: [state(0, 0, 0), state(0, 0, 3)]

Bước 3:

Trạng thái đang xét : state(0, 3, 0)

Trạng thái sinh ra: state(8, 3, 0), state(0, 0, 0), state(0, 5, 0), state(3, 0, 0), state(0, 0, 3), state(0, 3, 3)

OPEN: [state(8, 3, 0), state(3, 0, 0), state(0, 3, 3), state(3, 0, 0), state(8, 0, 3), state(0, 5, 3), state(8, 0, 0), state(0, 5, 0)]

CLOSED: [state(0, 0, 0), state(0, 0, 3), state(0, 3, 0)]

Bước 4:

Trạng thái đang xét : state(8, 3, 0)

Trạng thái sinh ra: state(6, 5, 0), state(0, 3, 0), state(8, 0, 0), state(8, 0, 3), state(5, 0, 3), state(8, 3, 3), state(8, 5, 0)

OPEN: [state(6, 5, 0), state(5, 0, 3), state(8, 3, 3), state(8, 5, 0), state(3, 0, 0), state(0, 3, 3), state(3, 0, 0), state(8, 0, 3), state(0, 5, 3), state(8, 0, 0), state(0, 5, 0)]

CLOSED: [state(0, 0, 0), state(0, 0, 3), state(0, 3, 0), state(8, 3, 0)]

Bước 5:

Trạng thái đang xét : state(6, 5, 0)

Trạng thái sinh ra: state(6, 0, 0), state(0, 5, 0), state(3, 5, 3), state(8, 5, 0), state(6, 5, 3), state(8, 3, 0), state(6, 2, 3)

OPEN: [state(6, 0, 0), state(3, 5, 3), state(6, 5, 3), state(6, 2, 3), state(8, 0, 3), state(5, 0, 3), state(8, 3, 3), state(8, 5, 0), state(3, 0, 0), state(0, 3, 3), state(3, 0, 0), state(8, 0, 3), state(0, 5, 3), state(8, 0, 0), state(0, 5, 0)]

CLOSED: [state(0, 0, 0), state(0, 0, 3), state(0, 3, 0), state(8, 3, 0), state(6, 5, 0)]

Bước 6:

Trạng thái đang xét : state(6, 0, 0)

Trạng thái sinh ra: state(6, 0, 3), state(0, 0, 0), state(1, 5, 3), state(3, 0, 3), state(6, 5, 0), state(8, 0, 0)

OPEN: [state(6, 0, 3), state(1, 5, 3), state(3, 0, 3), state(6, 5, 0), state(3, 5, 3), state(6, 5, 3), state(6, 2, 3), state(8, 0, 3), state(5, 0, 3), state(8, 3, 3), state(8, 5, 0), state(3, 0, 0), state(0, 3, 3), state(3, 0, 0), state(8, 0, 3), state(0, 5, 3), state(8, 0, 0), state(0, 5, 0)]

CLOSED: [state(0, 0, 0), state(0, 0, 3), state(0, 3, 0), state(8, 3, 0), state(6, 5, 0), state(6, 0, 0)]

Bước 7:

Trạng thái đang xét : state(6, 0, 3)

Trạng thái sinh ra: state(6, 3, 0), state(0, 0, 3), state(6, 0, 0), state(1, 5, 3), state(8, 0, 3)

OPEN: [state(6, 3, 0), state(0, 0, 3), state(1, 5, 3), state(3, 0, 3), state(6, 5, 0), state(3, 5, 3), state(6, 5, 3), state(6, 2, 3), state(8, 0, 3), state(5, 0, 3), state(8, 3, 3), state(8, 5, 0), state(3, 0, 0), state(0, 3, 3), state(3, 0, 0), state(8, 0, 3), state(0, 5, 3), state(8, 0, 0), state(0, 5, 0)]

CLOSED: [state(0, 0, 0), state(0, 0, 3), state(0, 3, 0), state(8, 3, 0), state(6, 5, 0), state(6, 0, 0), state(6, 0, 3)]

Bước 8:

Trạng thái đang xét : state(6, 3, 0)

Trạng thái sinh ra: state(4, 5, 0), state(8, 3, 0), state(6, 5, 0), state(0, 3, 0), state(6, 0, 3), state(6, 0, 0)

OPEN: [state(4, 5, 0), state(8, 3, 0), state(1, 5, 3), state(3, 0, 3), state(6, 5, 0), state(3, 5, 3), state(6, 5, 3), state(6, 2, 3), state(8, 0, 3), state(5, 0, 3), state(8, 3, 3), state(8, 5, 0), state(3, 0, 0), state(0, 3, 3), state(3, 0, 0), state(8, 0, 3), state(0, 5, 3), state(8, 0, 0), state(0, 5, 0)]

CLOSED: [state(0, 0, 0), state(0, 0, 3), state(0, 3, 0), state(8, 3, 0), state(6, 5, 0), state(6, 0, 0), state(6, 0, 3), state(6, 3, 0)]

Bước 9:

Trạng thái đang xét: state(4, 5, 0) (Đạt mục tiêu)

....

(Tiếp tục cho đến khi OPEN rỗng là đã vét hết nghiệm)

### **-Chương trình Prolog:**

% Khởi tạo dung tích tối đa của mỗi bình

max\_capacity(8, 5, 3).

% Trạng thái đích là trạng thái mà bình 1 chứa đúng 4 lít nước

goal(state(4, \_, \_)).

goal(state(\_, 4, \_)).

goal(state(\_, \_, 4)).

% Hàm kiểm tra trạng thái đã được duyệt chưa

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

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

% Các hành động đổ đầy, làm rỗng, và chuyển nước giữa các bình

move(state(A, B, C), state(8, B, C)) :- max\_capacity(8, \_, \_), A < 8. % Đổ đầy bình 1

move(state(A, B, C), state(A, 5, C)) :- max\_capacity(\_, 5, \_), B < 5. % Đổ đầy bình 2  
move(state(A, B, C), state(A, B, 3)) :- max\_capacity(\_, \_, 3), C < 3. % Đổ đầy bình 3

move(state(A, B, C), state(0, B, C)) :- A > 0. % Làm rỗng bình 1  
move(state(A, B, C), state(A, 0, C)) :- B > 0. % Làm rỗng bình 2  
move(state(A, B, C), state(A, B, 0)) :- C > 0. % Làm rỗng bình 3

% Chuyển nước từ bình 1 sang bình 2

move(state(A, B, C), state(A1, B1, C)) :-

A > 0, B < 5, T is min(A, 5 - B), A1 is A - T, B1 is B + T.

% Chuyển nước từ bình 1 sang bình 3

move(state(A, B, C), state(A1, B, C1)) :-

A > 0, C < 3, T is min(A, 3 - C), A1 is A - T, C1 is C + T.

% Chuyển nước từ bình 2 sang bình 3

move(state(A, B, C), state(A, B1, C1)) :-

B > 0, C < 3, T is min(B, 3 - C), B1 is B - T, C1 is C + T.

% Chuyển nước từ bình 3 sang bình 2

move(state(A, B, C), state(A, B1, C1)) :-

C > 0, B < 5, T is min(C, 5 - B), C1 is C - T, B1 is B + T.

% Chuyển nước từ bình 3 sang bình 1

move(state(A, B, C), state(A1, B, C1)) :-

C > 0, A < 8, T is min(C, 8 - A), C1 is C - T, A1 is A + T.

% Chuyển nước từ bình 2 sang bình 1

move(state(A, B, C), state(A1, B1, C)) :-

$B > 0, A < 8, T \text{ is } \min(B, 8 - A), B1 \text{ is } B - T, A1 \text{ is } A + T.$

% Thuật toán DFS tìm trạng thái đích

solve(State, Path) :- dfs([State], [], Path).

dfs([State|\_], \_, [State]) :- goal(State).

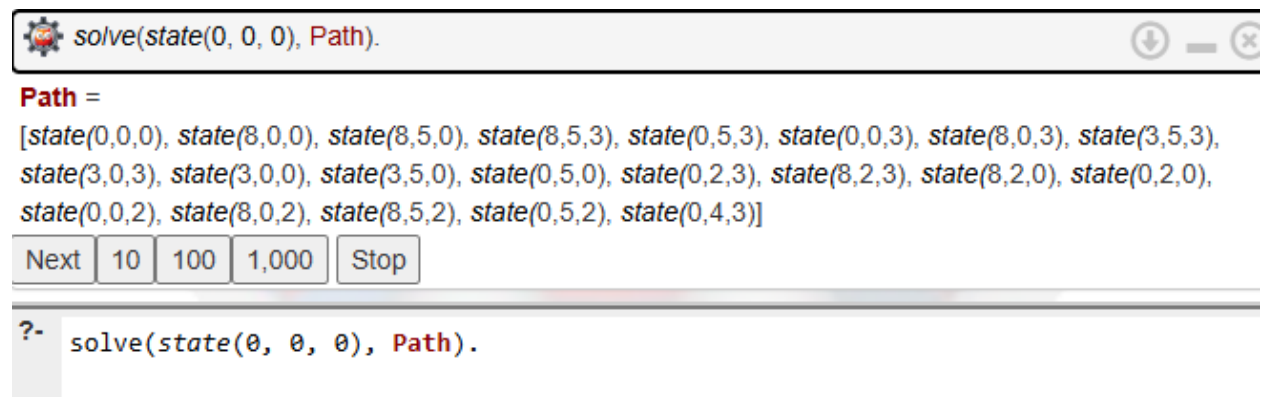
dfs([State|Rest], Closed, [State|Path]) :-

findall(Next, (move(State, Next), \+ member(Next, Closed)), NextStates),

append(NextStates, Rest, NewOpen),

dfs(NewOpen, [State|Closed], Path).

**Kết quả:**



**Path =**

[state(0,0,0), state(8,0,0), state(8,5,0), state(8,5,3), state(0,5,3), state(0,0,3), state(8,0,3), state(3,5,3), state(3,0,3), state(3,0,0), state(3,5,0), state(0,5,0), state(0,2,3), state(8,2,3), state(8,2,0), state(0,2,0), state(0,0,2), state(8,0,2), state(8,5,2), state(0,5,2), state(0,4,3)]

Next 10 100 1,000 Stop

?- solve(state(0, 0, 0), Path).