

TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI
TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

=====***=====



BÁO CÁO BTL THUỘC HỌC PHẦN
TRÍ TUỆ NHÂN TẠO

TÌM HIỂU THUẬT TOÁN TÌM KIẾM MÙ VÀ ỨNG
DỤNG VÀO BÀI TOÁN TIC TAC TOE

GVHD: ThS. Mai Thanh Hồng
Nhóm – Lớp: 12 - 20242IT6094003
Thành viên: Trần Đào Duy Anh - 2023600477
Trịnh Gia Khánh - 2023601012
Trương Công Minh - 2023600677
Trần Thị Thảo Ngân - 2023602673

Hà Nội, năm 2025

**KẾ HOẠCH THỰC HIỆN TIỂU LUẬN, BÀI TẬP LỚN, ĐỒ
ÁN/DỰ ÁN**

Tên lớp: 20242IT6094003

Khoá: 18

Tên nhóm: 12

Họ và tên thành viên trong nhóm:

(1)Họ và tên SV: Trần Đào Duy Anh

Mã SV: 2023600477

(2)Họ và tên SV: Trịnh Gia Khánh

Mã SV: 2023601012

(3)Họ và tên SV: Trương Công Minh

Mã SV: 2023600677

(4)Họ và tên SV: Trần Thị Thảo Ngân

Mã SV: 2023602673

Tên chủ đề: Tìm hiểu các thuật toán tìm kiếm mù và ứng dụng vào bài toán tic tac toe

Tuần	Người thực hiện	Nội dung công việc	Phương pháp thực hiện
1	Cả nhóm	Viết lời mở đầu, lời cảm ơn, trình bày về không gian trạng thái, thuật toán tìm kiếm mù	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
2	Cả nhóm	Trình bày về tìm kiếm tối ưu và các thuật toán AKT, AT, A*	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
3	Cả nhóm	Mô tả không gian trạng thái của bài toán được chọn và toán tử chuyển trạng thái	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
4	Cả nhóm	Khởi tạo các biến cần thiết và xây dựng thuật toán tìm kiếm DFS	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
5	Cả nhóm	Lập trình để giải quyết bài toán đề ra	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo

6	Cả nhóm	Hoàn thiện báo cáo và các biểu mẫu, phiếu liên quan	Tổng hợp tất cả các nội dung báo cáo, các biểu mẫu, phiếu liên quan
---	---------	---	---

Ngày tháng năm 2025

XÁC NHẬN CỦA GIẢNG VIÊN

(Ký, ghi rõ họ tên)

BÁO CÁO HỌC TẬP NHÓM

Tên lớp : 2024IT6094003

Khóa : 18

Tên nhóm : Nhóm 12

Họ và tên thành viên trong nhóm :

(1)Họ và tên SV: Trần Đào Duy Anh

Mã SV: 2023600477

(2)Họ và tên SV: Trịnh Gia Khánh

Mã SV: 2023601012

(3)Họ và tên SV: Trương Công Minh

Mã SV: 2023600677

(4)Họ và tên SV: Trần Thị Thảo Ngân

Mã SV: 2023602673

Tên chủ đề : Tìm hiểu các thuật toán tìm kiếm mù và ứng dụng vào bài toán tic tac toe

Tuần	Người thực hiện	Nội dung công việc	Kết quả đạt được	Kiến nghị với giảng viên hướng dẫn (Nêu những khó khăn, hỗ trợ từ phía giảng viên, ... nếu cần)
1	Cả nhóm	Viết lời cảm ơn, lời mở đầu, trình bày về không gian trạng thái và thuật toán tìm kiếm mù	Các thành viên hoàn thành đúng thời hạn và đầy đủ nội dung	
2	Cả nhóm	Trình bày về tìm kiếm tối ưu và các thuật toán AKT, AT, A*	Các thành viên hoàn thành đúng thời hạn và đầy đủ nội dung	
3	Cả nhóm	Mô tả không gian trạng thái của bài toán được chọn và toán tử chuyển trạng thái	Các thành viên hoàn thành đúng thời hạn và đầy đủ nội dung	
4	Cả nhóm	Khởi tạo các biến cần thiết và xây dựng thuật toán tìm kiếm DFS	Cả nhóm đã thống nhất về các biến và xây dựng được thuật toán	

5	Cả nhóm	Lập trình để giải quyết bài toán đề ra	Cả nhóm lựa chọn ngôn ngữ lập trình Python, sử dụng các biến và thuật toán đã xây dựng để hoàn thành sản phẩm. Kết quả chạy thử ra đúng đáp án	
6	Cả nhóm	Hoàn thiện báo cáo và các biểu mẫu, phiếu liên quan	Cả nhóm hoàn thành đầy đủ và đúng thời hạn	

Ngày tháng năm 2025

XÁC NHẬN CỦA GIẢNG VIÊN

(Ký, ghi rõ họ tên)

PHIẾU HỌC TẬP NHÓM

I. Thông tin chung

1. Tên lớp: 2024IT6094003 Khóa: 18
2. Tên nhóm: 12
3. Họ và tên thành viên trong nhóm:

(1)Họ và tên SV: Trần Đào Duy Anh	Mã SV: 2023600477
(2)Họ và tên SV: Trịnh Gia Khánh	Mã SV: 2023601012
(3)Họ và tên SV: Trương Công Minh	Mã SV: 2023600677
(4)Họ và tên SV: Trần Thị Thảo Ngân	Mã SV: 2023602673

II. Nội dung học tập

1. Tên chủ đề: Tìm hiểu các thuật toán tìm kiếm mù và ứng dụng vào bài toán tic tac toe
2. Hoạt động của sinh viên
 - *Hoạt động 1: Đề xuất chủ đề nghiên cứu*
 - + Nội dung:
 - Viết đề xuất lựa chọn chủ đề nghiên cứu và xin ý kiến người hướng dẫn về chủ đề nghiên cứu.
 - Lập biên bản họp và làm việc nhóm (BM03)
 - Đặt ra các quy tắc làm việc nhóm:
 - Các thành viên cần tham gia đầy đủ và đúng giờ các cuộc họp
 - Vắng mặt phải báo trước với nhóm trưởng trước buổi họp ít nhất 12 tiếng và có lý do chính đáng
 - Mỗi thành viên có trách nhiệm hoàn thành công việc đã được giao nghiêm túc và đúng hạn
 - Có vướng mắc thì trao đổi với nhóm trưởng và các thành viên trong nhóm qua nhóm Zalo. Nếu không tìm được hướng giải quyết thì hỏi giảng viên hướng dẫn – Cô Mai Thanh Hồng
 - + Mục tiêu/chuẩn đầu ra: Lập biên bản họp và làm việc nhóm (BM03)
 - *Hoạt động 2: Báo cáo tiến độ lần 1*
 - + Mục tiêu/chuẩn đầu ra: Viết được nội dung phần Mở đầu và Chương 1: Không gian trạng thái và các thuật toán tìm kiếm: Giới thiệu tổng quan về không gian trạng thái, toán tử chuyển trạng thái, tìm hiểu về thuật toán tìm kiếm theo chiều sâu (DFS), thuật toán tìm kiếm theo chiều rộng (BFS), thuật toán Heuristic.

- *Hoạt động 3: Báo cáo tiến độ lần 2*
 - + Mục tiêu/chuẩn đầu ra: Viết được nội dung Chương 2: Xây dựng chương trình: Tiến hành mô tả bài toán đã được đưa ra và sử dụng ngôn ngữ lập trình Python để ứng dụng thuật toán tìm kiếm mù giải bài toán tic tac toe.
- *Hoạt động 4: Nộp cuốn Báo cáo thí nghiệm/thực nghiệm và chương trình code*
 - + Mục tiêu/chuẩn đầu ra: Hoàn thiện và nộp sản phẩm nghiên cứu
- 3. Sản phẩm nghiên cứu: Quyển báo cáo thí nghiệm/thực nghiệm + Chương trình code

III. Nhiệm vụ học tập

1. Hoàn thành báo cáo thí nghiệm/thực nghiệm theo đúng thời gian quy định (từ ngày 21/04/2025 đến ngày 25/05/2025)
2. Báo cáo sản phẩm nghiên cứu theo chủ đề được giao khi kết thúc thời gian thực hiện.

IV. Học liệu thực hiện Bài tập lớn

1. Tài liệu học tập

[1] Nguyễn Phương Nga, Trần Hùng Cường, *Giáo trình Trí tuệ nhân tạo*, NXB Thống kê, 2021.

[2] Nguyễn Thanh Thủy, *Trí tuệ nhân tạo*, NXB Thống kê, 1999.

[3] Nguyễn Đình Thức, *Giáo trình Trí tuệ nhân tạo: Mạng neuron phương pháp và ứng dụng*, NXB Giáo dục, 2000.

2. Phương tiện, nguyên liệu thực hiện báo cáo: Máy tính cá nhân, máy chiếu, mạng internet.

KẾ HOẠCH LÀM VIỆC NHÓM								
1. Tên lớp: 20242IT6094003 Khóa: 18								
2. Nhóm: 12								
3. Ngày bắt đầu: 21/04/2025								
4. Ngày kết thúc: 25/05/2025								
5. Thành viên nhóm:								
(1)Họ và tên SV: Trần Đào Duy Anh						Mã SV: 2023600477		
(2)Họ và tên SV: Trịnh Gia Khánh						Mã SV: 2023601012		
(3)Họ và tên SV: Trương Công Minh						Mã SV: 2023600677		
(4)Họ và tên SV: Trần Thị Thảo Ngân						Mã SV: 2023602673		
#	Công việc	Ngày bắt đầu dự kiến	Ngày bắt đầu thực tế	Ngày kết thúc dự kiến	Ngày kết thúc thực tế	Trạng thái	Người thực hiện	Ghi chú
1	Viết lời mở đầu và lời cảm ơn	21/04/2025	21/04/2025	25/04/2025	25/04/2025	Done	Trịnh Gia Khánh	
2	Trình bày về không gian trạng thái	21/04/2025	21/04/2025	25/04/2025	25/04/2025	Done	Trịnh Gia Khánh & Trần Thị Thảo Ngân	
3	Trình bày về thuật toán tìm	21/04/2025	21/04/2025	25/04/2025	25/04/2025	Done	Trương Công Minh & Trần Đào Duy Anh	

	kiểm mù							
4	Trình bày về tìm kiếm tối ưu	28/04 /2025	28/04 /2025	02/05 /2025	02/05 /2025	Done	Trần Đào Duy Anh	
5	Trình bày về thuật toán AT	28/04 /2025	28/04 /2025	02/05 /2025	02/05 /2025	Done	Trịnh Gia Khánh	
6	Trình bày về thuật toán AKT	28/04 /2025	28/04 /2025	02/05 /2025	02/05 /2025	Done	Trương Công Minh	
7	Trình bày về thuật toán A*	28/04 /2025	28/04 /2025	02/05 /2025	02/05 /2025	Done	Trần Thị Thảo Ngân	
8	Mô tả không gian trạng thái của bài toán được chọn	05/05 /2025	05/05 /2025	09/05 /2025	09/05 /2025	Done	Cả nhóm	
9	Toán tử chuyển	05/05 /2025	05/05 /2025	09/05 /2025	09/05 /2025	Done	Cả nhóm	

	trạng thái							
10	Khởi tạo các biến cần thiết và xây dựng thuật toán tìm kiếm DFS	12/05 /2025	12/05 /2025	16/05 /2025	16/05 /2025	Done	Cả nhóm	
11	Lập trình để giải quyết bài toán đề ra	19/05 /2025	19/05 /2025	23/05 /2025	23/05 /2025	Done	Cả nhóm	
12	Hoàn thiện báo cáo và các biểu mẫu, phiếu liên quan	24/05 /2025	24/05 /2025	25/05 /2025	25/05 /2025	Done	Cả nhóm	

BIÊN BẢN HỌP, LÀM VIỆC NHÓM TUẦN 1					
1. Tên lớp: 20242IT6094003 Khóa: 18					
2. Nhóm: 12					
3. Thời gian: 21/04/2025					
4. Địa điểm: Google meet					
5. Người chủ trì cuộc họp: Trương Công Minh					
6. Thành viên tham dự - Participants: Trịnh Gia Khánh, Trần Đào Duy Anh, Trần Thị Thảo Ngân					
#	Thành viên	Đúng giờ - On time	Muộn - Late	Vắng - Absent	Ghi chú
1	Trần Đào Duy Anh	✓			Done
2	Trịnh Gia Khánh	✓			Done
3	Trương Công Minh	✓			Done
4	Trần Thị Thảo Ngân	✓			Done
Chương trình họp - Meeting agenda					
#	Mục nội dung - Items	Người trình bày - Owner(s)	Thời gian - Time	Ghi chú, trao đổi - Note	
1	Quản lý nhóm	Cả nhóm	15p	Bầu trưởng nhóm cho mỗi lần	
2	Thiết lập kênh giao tiếp, lưu trữ	Trương Công Minh	5p	- Kênh giao tiếp của cả nhóm thông qua: Zalo - Kênh lưu trữ: Google drive	
3	Thống nhất quy định và mục tiêu làm việc nhóm	Cả nhóm	15p	- Luân phiên làm trưởng nhóm cho từng buổi họp. - Hoàn thành bài tập đúng hạn - Xây dựng quy tắc chung cho nhóm: 1. Đảm bảo tham dự đầy đủ và đúng giờ các cuộc họp.	

				2. Chấp nhận vắng mặt khi có lý do xác đáng. 3. Đề cao trách nhiệm hoàn thành các công việc đã được giao phó.
4	Xác định đề tài	Cả nhóm	30p	Thống nhất đề tài làm bài tập lớn
5	Phân chia công việc cho từng thành viên	Trương Công Minh	10p	- Viết lời mở đầu, Lời cảm ơn: Trịnh Gia Khánh - Trình bày về không gian trạng thái: Trần Thị Thảo Ngân & Trịnh Gia Khánh - Trình bày về thuật toán tìm kiếm mù: Trương Công Minh & Trần Đào Duy Anh

Vấn đề & Giải pháp - Issues/problems & Solutions

#	Vấn đề - Issues/problems	Các giải pháp đề xuất - Suggested solutions	Giải pháp được chọn - Selected solution	Ghi chú - Notes
1	Xác định đề tài	Tìm hiểu thuật toán tìm kiếm mù và ứng dụng vào thuật toán tic tac toe	Tìm hiểu thuật toán tìm kiếm	Ứng dụng vào thuật toán tic tac toe
2	Xác định ngôn ngữ lập trình	Cân nhắc lựa chọn ngôn ngữ lập trình đã học, dễ hiểu, dễ tiếp thu	Ưu tiên lựa chọn ngôn ngữ lập trình đã học và dễ hiểu	

Kế hoạch hoạt động - Action plan

#	Hoạt động - Action	Thời hạn - Deadline	Người thực hiện - Owner(s)	Ghi chú, trao đổi - Notes
---	--------------------	------------------------	-------------------------------	---------------------------

1	Viết lời cảm ơn và lời mở đầu	25/04/2025	Trịnh Gia Khánh	Hoàn thành việc viết lời cảm ơn và lời mở đầu	
2	Trình bày về không gian trạng thái	25/04/2025	Trần Thị Thảo Ngân & Trịnh Gia Khánh	Hoàn thành việc trình bày về không gian trạng thái	
3	Trình bày về thuật toán tìm kiếm mù	25/04/2025	Trương Công Minh &Trần Đào Duy Anh	Hoàn thành việc trình bày về thuật toán tìm kiếm mù	
Đóng góp nhóm - Team contribution					
#	Thành viên - Member	Ý tưởng, giải pháp - Ideas	Hỗ trợ người khác - Support other(s)	Hoạt động xây dựng nhóm - Team building activities	Ghi chú - Notes
1	Trần Đào Duy Anh	2	0	1	
2	Trịnh Gia Khánh	2	1	1	
3	Trương Công Minh	3	0	1	
4	Trần Thị Thảo Ngân	2	1	1	
Kết quả đánh giá phản hồi của nhóm - Team feedback					
#	Số phiếu 4	Số phiếu 3	Số phiếu 2	Số phiếu 1	
1	4	0	0	0	

BIÊN BẢN HỌP, LÀM VIỆC NHÓM TUẦN 2					
1. Tên lớp: 20242IT6094003 Khóa: 18					
2. Nhóm: 12					
3. Thời gian: 28/04/2025					
4. Địa điểm: Google meet					
5. Người chủ trì cuộc họp: Trần Thị Thảo Ngân					
6. Thành viên tham dự - Participants: Trần Đào Duy Anh, Trịnh Gia Khánh, Trương Công Minh					
#	Thành viên	Đúng giờ - On time	Muộn - Late	Vắng - Absent	Ghi chú
1	Trần Đào Duy Anh	✓			Done
2	Trịnh Gia Khánh	✓			Done
3	Trương Công Minh	✓			Done
4	Trần Thị Thảo Ngân	✓			Done
Chương trình họp - Meeting agenda					
#	Mục nội dung - Items	Người trình bày - Owner(s)	Thời gian - Time	Ghi chú, trao đổi - Note	
1	Check tiến độ báo cáo	Trần Thị Thảo Ngân	30p	Tất cả hoàn thành nhiệm vụ đúng hạn	
2	Phân chia công việc cho từng thành viên	Trần Thị Thảo Ngân	30p	- Trình bày về tìm kiếm tối ưu (Trần Đào Duy Anh) - Trình bày về thuật toán AT (Trịnh Gia Khánh) - Trình bày về thuật toán AKT (Trương Công Minh) - Trình bày về thuật toán A* (Trần Thị Thảo Ngân)	
Vấn đề & Giải pháp - Issues/problems & Solutions					
#	Vấn đề - Issues/problems	Các giải pháp đề xuất - Suggested solutions		Giải pháp được chọn -	Ghi chú - Notes

			Selected solution	
1	Chưa tìm được cách áp dụng thuật toán cho bài toán	Tham khảo, tìm hiểu chương trình có sẵn, xem các tài liệu trên học kết hợp	Tham khảo, tìm hiểu tài liệu, chương trình có sẵn	

Kế hoạch hoạt động - Action plan

#	Hoạt động - Action	Thời hạn - Deadline	Người thực hiện - Owner(s)	Ghi chú, trao đổi - Notes
1	Trình bày về tìm kiếm tối ưu	02/05/2025	Trần Đào Duy Anh	Hoàn thành việc trình bày về tìm kiếm tối ưu
2	Trình bày về thuật toán AT	02/05/2025	Trịnh Gia Khánh	Hoàn thành việc trình bày về thuật toán AT
3	Trình bày về thuật toán AKT	02/05/2025	Trương Công Minh	Hoàn thành việc trình bày về thuật toán AKT
4	Trình bày về thuật toán A*	02/05/2025	Trần Thị Thảo Ngân	Hoàn thành việc trình bày về thuật toán A*

Đóng góp nhóm - Team contribution

#	Thành viên - Member	Ý tưởng, giải pháp - Ideas	Hỗ trợ người khác - Support other(s)	Hoạt động xây dựng nhóm - Team building activities	Ghi chú - Notes
1	Trần Đào Duy Anh	1	1	1	
2	Trịnh Gia Khánh	1	2	2	
3	Trương Công Minh	1	1	1	
4	Trần Thị Thảo Ngân	1	1	1	

Kết quả đánh giá phản hồi của nhóm - Team feedback

#	Số phiếu 4	Số phiếu 3	Số phiếu 2	Số phiếu 1	
1	4	0	0	0	

BIÊN BẢN HỌP, LÀM VIỆC NHÓM TUẦN 3					
1. Tên lớp: 20242IT6094003 Khóa: 18					
2. Nhóm: 12					
3. Thời gian: 05/05/2025					
4. Địa điểm: Google meet					
5. Người chủ trì cuộc họp: Trịnh Gia Khánh					
6. Thành viên tham dự - Participants: Trần Đào Duy Anh, Trương Công Minh, Trần Thị Thảo Ngân					
#	Thành viên	Đúng giờ - On time	Muộn - Late	Vắng - Absent	Ghi chú
1	Trần Đào Duy Anh	✓			Done
2	Trịnh Gia Khánh	✓			Done
3	Trương Công Minh	✓			Done
4	Trần Thị Thảo Ngân	✓			Done
Chương trình họp - Meeting agenda					
#	Mục nội dung - Items	Người trình bày - Owner(s)	Thời gian - Time	Ghi chú, trao đổi - Note	
1	Check tiến độ báo cáo	Trịnh Gia Khánh	30p	Tất cả hoàn thành nhiệm vụ đúng hạn	
2	Trao đổi chương 2: Xây dựng chương trình phân mô tả bài toán	Cả nhóm	40p	- Mô tả không gian trạng thái của bài toán được chọn - Toán tử chuyển trạng thái	
Vấn đề & Giải pháp - Issues/problems & Solutions					
#	Vấn đề - Issues/problems	Các giải pháp đề xuất - Suggested solutions		Giải pháp được chọn - Selected solution	Ghi chú - Notes
1	Đưa ra bài toán cụ thể để áp dụng thuật toán	Tham khảo các bài toán trong giáo trình		Chọn bài toán: Bảng 3x3, hai người chơi thay	NT: Người đầu tiên có 3 dấu giống nhau

			phiên đánh “X” và “O”.	theo hàng, cột hoặc đường chéo sẽ thắng. Nếu bảng đầy mà không ai thắng thì hòa
--	--	--	------------------------	---

Kế hoạch hoạt động - Action plan

#	Hoạt động - Action	Thời hạn - Deadline	Người thực hiện - Owner(s)	Ghi chú, trao đổi - Notes
1	Mô tả không gian trạng thái của bài toán được chọn	09/05/2025	Cả nhóm	Hoàn thành việc mô tả không gian trạng thái của bài toán được chọn
2	Toán tử chuyển trạng thái	09/05/2025	Cả nhóm	Hoàn thành việc mô tả toán tử chuyển trạng thái

Đóng góp nhóm - Team contribution

#	Thành viên - Member	Ý tưởng, giải pháp - Ideas	Hỗ trợ người khác - Support other(s)	Hoạt động xây dựng nhóm - Team building activities	Ghi chú - Notes
1	Trần Đào Duy Anh	1	0	2	
2	Trịnh Gia Khánh	2	1	0	
3	Trương Công Minh	1	1	1	
4	Trần Thị Thảo Ngân	1	1	1	

Kết quả đánh giá phản hồi của nhóm - Team feedback

#	Số phiếu 4	Số phiếu 3	Số phiếu 2	Số phiếu 1	
1	4	0	0	0	

BIÊN BẢN HỌP, LÀM VIỆC NHÓM TUẦN 4					
1. Tên lớp: 20242IT6094003 Khóa: 18					
2. Nhóm: 12					
3. Thời gian: 12/05/2025					
4. Địa điểm: Google meet					
5. Người chủ trì cuộc họp: Trần Đào Duy Anh					
6. Thành viên tham dự - Participants: Trịnh Gia Khánh, Trương Công Minh, Trần Thị Thảo Ngân					
#	Thành viên	Đúng giờ - On time	Muộn - Late	Vắng - Absent	Ghi chú
1	Trần Đào Duy Anh	✓			Done
2	Trịnh Gia Khánh	✓			Done
3	Trương Công Minh	✓			Done
4	Trần Thị Thảo Ngân	✓			Done
Chương trình họp - Meeting agenda					
#	Mục nội dung - Items	Người trình bày - Owner(s)	Thời gian - Time	Ghi chú, trao đổi - Note	
1	Check tiến độ báo cáo	Trần Đào Duy Anh	30p	Tất cả hoàn thành nhiệm vụ đúng hạn	
2	Trao đổi về vấn đề sử dụng ngôn ngữ lập trình để áp dụng thuật toán với đề bài đã đưa ra	Cả nhóm	30p	- Sử dụng ngôn ngữ Python	
Vấn đề & Giải pháp - Issues/problems & Solutions					
#	Vấn đề - Issues/problems	Các giải pháp đề xuất - Suggested solutions		Giải pháp được chọn - Selected solution	Ghi chú - Notes
Kế hoạch hoạt động - Action plan					

#	Hoạt động - Action	Thời hạn - Deadline	Người thực hiện - Owner(s)	Ghi chú, trao đổi - Notes	
1	Khởi tạo các biến cần thiết và xây dựng thuật toán tìm kiếm DFS	16/05/2025	Cả nhóm	Hoàn thành việc khởi tạo các biến cần thiết và xây dựng thuật toán tìm kiếm DFS	
Đóng góp nhóm - Team contribution					
#	Thành viên - Member	Ý tưởng, giải pháp - Ideas	Hỗ trợ người khác - Support other(s)	Hoạt động xây dựng nhóm - Team building activities	Ghi chú - Notes
1	Trần Đào Duy Anh	1	0	1	
2	Trịnh Gia Khánh	1	1	0	
3	Trương Công Minh	0	1	1	
4	Trần Thị Thảo Ngân	1	0	1	
Kết quả đánh giá phản hồi của nhóm - Team feedback					
#	Số phiếu 4	Số phiếu 3	Số phiếu 2	Số phiếu 1	
1	4	0	0	0	

BIÊN BẢN HỌP, LÀM VIỆC NHÓM TUẦN 5					
1. Tên lớp: 20242IT6094003 Khóa: 18					
2. Nhóm: 12					
3. Thời gian: 19/05/2025					
4. Địa điểm: Google meet					
5. Người chủ trì cuộc họp: Trương Công Minh					
6. Thành viên tham dự - Participants: Trần Đào Duy Anh, Trịnh Gia Khánh, Trần Thị Thảo Ngân					
#	Thành viên	Đúng giờ - On time	Muộn - Late	Vắng - Absent	Ghi chú
1	Trần Đào Duy Anh	✓			Done
2	Trịnh Gia Khánh	✓			Done
3	Trương Công Minh	✓			Done
4	Trần Thị Thảo Ngân	✓			Done
Chương trình họp - Meeting agenda					
#	Mục nội dung - Items	Người trình bày - Owner(s)	Thời gian - Time	Ghi chú, trao đổi - Note	
1	Check tiến độ báo cáo	Trương Công Minh	30p	Tất cả hoàn thành nhiệm vụ đúng hạn	
2	Lập trình để giải quyết bài toán đề ra	Cả nhóm	30p	- Sử dụng ngôn ngữ Python. Code trên Google Colaboratory	
Vấn đề & Giải pháp - Issues/problems & Solutions					
#	Vấn đề - Issues/problems	Các giải pháp đề xuất - Suggested solutions		Giải pháp được chọn - Selected solution	Ghi chú - Notes
Kế hoạch hoạt động - Action plan					

#	Hoạt động - Action	Thời hạn - Deadline	Người thực hiện - Owner(s)	Ghi chú, trao đổi - Notes	
1	Lập trình để giải quyết bài toán đề ra	23/05/2025	Cả nhóm	Hoàn thành việc lập trình để giải quyết bài toán đề ra	
Đóng góp nhóm - Team contribution					
#	Thành viên - Member	Ý tưởng, giải pháp - Ideas	Hỗ trợ người khác - Support other(s)	Hoạt động xây dựng nhóm - Team building activities	Ghi chú - Notes
1	Trần Đào Duy Anh	1	1	0	
2	Trịnh Gia Khánh	0	1	1	
3	Trương Công Minh	0	1	1	
4	Trần Thị Thảo Ngân	1	0	1	
Kết quả đánh giá phản hồi của nhóm - Team feedback					
#	Số phiếu 4	Số phiếu 3	Số phiếu 2	Số phiếu 1	
1	4	0	0	0	

BIÊN BẢN HỌP, LÀM VIỆC NHÓM TUẦN 6					
1. Tên lớp: 20242IT6094003 Khóa: 18					
2. Nhóm: 12					
3. Thời gian: 24/05/2025					
4. Địa điểm: Google meet					
5. Người chủ trì cuộc họp: Trần Thị Thảo Ngân					
6. Thành viên tham dự - Participants: Trần Đào Duy Anh, Trịnh Gia Khánh, Trương Công Minh					
#	Thành viên	Đúng giờ - On time	Muộn - Late	Vắng - Absent	Ghi chú
1	Trần Đào Duy Anh	✓			Done
2	Trịnh Gia Khánh	✓			Done
3	Trương Công Minh	✓			Done
4	Trần Thị Thảo Ngân	✓			Done
Chương trình họp - Meeting agenda					
#	Mục nội dung - Items	Người trình bày - Owner(s)	Thời gian - Time	Ghi chú, trao đổi - Note	
1	Check tiến độ báo cáo	Trần Thị Thảo Ngân	30p	Tất cả hoàn thành nhiệm vụ đúng hạn	
2	Hoàn thiện báo cáo	Cả nhóm	30p		
Vấn đề & Giải pháp - Issues/problems & Solutions					
#	Vấn đề - Issues/problems	Các giải pháp đề xuất - Suggested solutions		Giải pháp được chọn - Selected solution	Ghi chú - Notes
Kế hoạch hoạt động - Action plan					
#	Hoạt động - Action	Thời hạn - Deadline	Người thực hiện - Owner(s)	Ghi chú, trao đổi - Notes	

1	Hoàn thiện báo cáo	25/05/2025	Cả nhóm		
Đóng góp nhóm - Team contribution					
#	Thành viên - Member	Ý tưởng, giải pháp - Ideas	Hỗ trợ người khác - Support other(s)	Hoạt động xây dựng nhóm - Team building activities	Ghi chú - Notes
1	Trần Đào Duy Anh	1	0	0	
2	Trịnh Gia Khánh	1	0	0	
3	Trương Công Minh	1	0	0	
4	Trần Thị Thảo Ngân	1	0	0	
Kết quả đánh giá phản hồi của nhóm - Team feedback					
#	Số phiếu 4	Số phiếu 3	Số phiếu 2	Số phiếu 1	
1	4	0	0	0	

LỜI CẢM ƠN

Lời đầu tiên, chúng em xin được bày tỏ lòng biết ơn sâu sắc đến Trường Đại học Công nghiệp Hà Nội đã tạo điều kiện cho sinh viên được tiếp cận với môn học "*Trí tuệ nhân tạo*" – một học phần hiện đại, thiết thực và mang tính ứng dụng cao trong thời đại công nghệ phát triển mạnh mẽ như hiện nay. Chúng em cũng xin gửi lời cảm ơn chân thành nhất đến quý thầy cô trong Trường Công nghệ thông tin và Truyền thông – những người đã tận tụy giảng dạy và truyền đạt những kiến thức quý báu, giúp chúng em từng bước hoàn thiện bản thân trên con đường học tập và rèn luyện.

Hơn hết, chúng em xin bày tỏ lòng kính trọng và biết ơn sâu sắc đến cô Mai Thanh Hồng, giảng viên trực tiếp giảng dạy học phần "*Trí tuệ nhân tạo*". Nhờ sự tận tâm, nhiệt huyết cùng phương pháp giảng dạy gần gũi, dễ hiểu của cô, đã giúp chúng em không chỉ tiếp thu được lượng kiến thức chuyên môn phong phú, mà còn được truyền cảm hứng và động lực để chủ động tìm tòi, học hỏi, phát triển tư duy sáng tạo trong lĩnh vực công nghệ ngày một hiện đại.

Tuy nhiên, với vốn kiến thức và kinh nghiệm thực tế còn hạn chế, dù đã nỗ lực hoàn thành bài báo cáo với tinh thần nghiêm túc và cầu thị, chúng em hiểu rằng không thể tránh khỏi những thiếu sót nhất định. Kính mong cô dành thời gian xem xét và đóng góp ý kiến để chúng em có cơ hội hoàn thiện bài báo cáo tốt hơn, đồng thời rút kinh nghiệm cho những lần sau.

Chúng em xin chân thành cảm ơn!

MỤC LỤC

DANH MỤC HÌNH ẢNH	1
DANH MỤC BẢNG BIỂU	2
LỜI MỞ ĐẦU	3
CHƯƠNG 1. KHÔNG GIAN TRẠNG THÁI VÀ CÁC THUẬT TOÁN TÌM KIẾM.....	4
1.1. Không gian trạng thái	4
1.1.1. Mô tả trạng thái.....	4
1.1.2. Toán tử chuyển trạng thái	4
1.1.3. Không gian trạng thái của bài toán	5
1.2. Các thuật toán tìm kiếm mù.....	5
1.2.1. Thuật toán tìm kiếm theo chiều sâu (Depth First Search)	5
1.2.2. Thuật toán tìm kiếm theo chiều rộng (Breadth First Search)	7
1.3. Thuật toán Heuristic.....	9
1.3.1. Tìm kiếm tối ưu (Best-First-Search).....	11
1.3.2. Thuật toán A^T	13
1.3.3. Thuật toán AKT	15
1.3.4. Thuật giải A^*	17
CHƯƠNG 2. XÂY DỰNG CHƯƠNG TRÌNH.....	23
2.1. Giới thiệu bài toán tic tac toe.....	23
2.2. Mô hình hóa bài toán	23
2.2.1. Không gian trạng thái	23
2.2.2. Toán tử chuyển trạng thái	24
2.3. Cài đặt thuật toán DFS cho tic tac toe	28
2.3.1. Ý tưởng sử dụng DFS	28
2.3.2. Cài đặt thuật toán.....	29
2.4. Kết quả.....	34
KẾT LUẬN	36
TÀI LIỆU THAM KHẢO.....	37

DANH MỤC HÌNH ẢNH

Hình 1.1. Ví dụ thuật toán DFS.....	6
Hình 1.2. Ví dụ thuật toán BFS.....	9
Hình 1.3. Ví dụ tìm kiếm tối ưu	11
Hình 1.4. Ví dụ thuật toán AT	14
Hình 1.5. Ví dụ thuật giải AKT	17
Hình 1.6. Ví dụ thuật giải A^*	20
Hình 2.1. Mô tả không gian trạng thái bài toán tic tac toe.....	24
Hình 2.2. Trạng thái đầu của bài toán	24
Hình 2.3. Trạng thái đích của bài toán	24
Hình 2.4. Cây trạng thái trò chơi Tic Tac Toe sử dụng thuật toán DFS.....	29
Hình 2.5. Thuật toán hiển thị bàn cờ.....	29
Hình 2.6. Kiểm tra chiến thắng	30
Hình 2.7. Kiểm tra hòa cờ	30
Hình 2.8. Tìm nước đi tối ưu.....	31
Hình 2.9. AI chọn nước đi tối ưu	32
Hình 2.10. Xử lý nước đi người chơi.....	32
Hình 2.11. Hàm main	33
Hình 2.12. Kết quả chạy thử	34
Hình 2.13. Kết quả chạy thử	34
Hình 2.14. Kết quả chạy thử	35

DANH MỤC BẢNG BIỂU

Bảng 1.1. Ví dụ thuật giải DFS	6
Bảng 1. 2: Ví dụ thuật giải BFS	9
Bảng 1.3. Ví dụ thuật giải AT.....	14
Bảng 1.4. So sánh thuật toán DFS và BFS.....	22

LỜI MỞ ĐẦU

Trong lĩnh vực trí tuệ nhân tạo và khoa học máy tính, việc giải quyết các bài toán tìm kiếm là một trong những chủ đề trọng tâm nhằm xây dựng hệ thống có khả năng tự động ra quyết định. Trong số đó, thuật toán tìm kiếm mù đóng vai trò nền tảng trong việc xây dựng các chiến lược giải bài toán mà không cần đến thông tin bổ sung về môi trường hay hướng dẫn từ đầu. Dù không tối ưu về hiệu suất so với các thuật toán tìm kiếm có thông tin, tìm kiếm mù vẫn là lựa chọn hiệu quả cho nhiều bài toán nhỏ và có không gian trạng thái hạn chế.

Trong số các thuật toán tìm kiếm mù, thuật toán tìm kiếm theo chiều sâu (Depth-First Search – DFS) là một phương pháp đơn giản nhưng hiệu quả trong việc khám phá không gian trạng thái của các bài toán có cấu trúc rõ ràng. DFS hoạt động dựa trên nguyên tắc đi sâu vào từng nhánh của cây tìm kiếm trước khi quay lại và mở rộng các nhánh khác, nhờ đó tiết kiệm bộ nhớ và phù hợp với các bài toán có độ sâu hữu hạn.

Từ đó, nhóm chúng em chọn ứng dụng **thuật toán DFS để giải bài toán Tic Tac Toe** – một trò chơi chiến lược đơn giản trên lưới 3x3 giữa hai người chơi. Mặc dù luật chơi đơn giản, Tic Tac Toe vẫn đòi hỏi chiến lược hợp lý để đảm bảo kết quả tối ưu. Bằng cách mô hình hóa trò chơi như một cây trạng thái và áp dụng thuật toán DFS, chương trình có thể tìm kiếm các nước đi hợp lý nhằm tối đa hóa khả năng thắng hoặc hòa.

CHƯƠNG 1. KHÔNG GIAN TRẠNG THÁI VÀ CÁC THUẬT TOÁN TÌM KIẾM

1.1. Không gian trạng thái

1.1.1. Mô tả trạng thái

Giải bài toán trong không gian trạng thái, trước hết phải xác định dạng mô tả trạng thái bài toán sao cho bài toán trở nên đơn giản hơn, phù hợp bản chất vật lý của bài toán (Có thể sử dụng các ký hiệu, vectơ, mảng hai chiều, cây, danh sách...).

Mỗi trạng thái chính là mỗi hình trạng của bài toán, các tình trạng ban đầu và tình trạng cuối của bài toán gọi là trạng thái đầu và trạng thái cuối.

Ví dụ: Bài toán đóng nước

Cho 2 bình có dung tích lần lượt là m và n (lit). Với nguồn nước không hạn chế, dùng 2 bình trên để đóng k lit nước. Không mất tính tổng quát có thể giả thiết $k \leq \min(m, n)$.

- Tại mỗi thời điểm xác định, lượng nước hiện có trong mỗi bình phản ánh bản chất hình trạng của bài toán ở thời điểm đó.
- Gọi x là lượng nước hiện có trong bình dung tích m và y là lượng nước hiện có trong bình dung tích n .
- Như vậy bộ có thứ tự (x, y) có thể xem là trạng thái của bài toán. Với cách mô tả như vậy, các trạng thái đặc biệt của bài toán sẽ là:

+ Trạng thái đầu: $(0,0)$

+ Trạng thái cuối: (x, k) hoặc (k, y)

1.1.2. Toán tử chuyển trạng thái

Toán tử chuyển trạng thái thực chất là các phép biến đổi đưa từ trạng thái này sang trạng thái khác.

Có hai cách dùng để biểu diễn các toán tử:

- Biểu diễn như một hàm xác định trên tập các trạng thái và nhận giá trị cũng trong tập này.
- Biểu diễn dưới dạng các quy tắc sản xuất S, A có nghĩa là nếu có trạng thái S thì có thể đưa đến trạng thái A .

Ví dụ 1. Bài toán đóng nước

Các thao tác sử dụng để chuyển trạng thái này sang trạng thái khác gồm:

- Đổ đầy một bình
- Đổ hết nước trong một bình ra ngoài

- Đổ nước từ bình này sang bình khác.

Như vậy, nếu trạng thái đang xét là (x, y) thì các trạng thái kế tiếp có thể chuyển đến sẽ là:

$$(x,y) \left\{ \begin{array}{l} (m, y) \\ (x, n) \\ (0, y) \\ (x, 0) \\ (0, x + y) \text{ nếu } x + y \leq n \\ (x + y - n, n) \text{ nếu } x + y > n \\ (x + y, 0) \text{ nếu } x + y \leq m \\ (m, x + y - m) \text{ nếu } x + y > m \end{array} \right.$$

1.1.3. Không gian trạng thái của bài toán

Không gian trạng thái là tập tất cả các trạng thái có thể có và tập các toán tử của bài toán.

Không gian trạng thái là một bộ bốn, ký hiệu: $K = (T, S, G, F)$. Trong đó:

- + T: tập tất cả các trạng thái có thể có của bài toán.
- + S: trạng thái đầu.
- + G: tập các trạng thái đích.
- + F: tập các toán tử

Ví dụ 1. Không gian trạng thái của bài toán đóng nước là bộ bốn T, S, G, F xác định như sau:

$$T = \{(x, y) / 0 \leq x \leq m; 0 \leq y \leq n\}$$

$$S = (0,0)$$

$$G = (x, k) \text{ hoặc } (k, y) / 0 \leq x \leq m; 0 \leq y \leq n\}$$

F = Tập các thao tác đóng đầy, đổ ra hoặc đổ sang bình khác thực hiện trên một bình.

1.2. Các thuật toán tìm kiếm mù

1.2.1. Thuật toán tìm kiếm theo chiều sâu (Depth First Search)

1.2.1.1. Tư tưởng của chiến lược tìm kiếm theo chiều sâu

- Từ đỉnh xuất phát duyệt một đỉnh kề.
- Các đỉnh của đồ thị được duyệt theo các nhánh đến nút lá.
- Nếu chưa tìm thấy đỉnh T_G thì quay lui tới một đỉnh nào đó để sang nhánh khác.
- Việc tìm kiếm kết thúc khi tìm thấy đỉnh T_G hoặc đã hết các đỉnh

1.2.1.2. Thuật toán tìm kiếm theo chiều sâu

Lưu trữ: Sử dụng hai danh sách DONG và MO trong đó:

DONG: Chứa các đỉnh đã xét, hoạt động theo kiểu FIFO (hàng đợi).

MO: chứa các đỉnh đang xét, hoạt động theo kiểu LIFO (ngăn xếp).

1. $MO = \emptyset$; $MO = MO \cup \{T_0\}$

2. while ($MO \neq \emptyset$)

{ $n = \text{get}(MO)$ // lấy đỉnh đầu trong danh sách MO

if ($n == T_G$) // nếu n là trạng thái kết thúc

return TRUE // tìm kiếm thành công, dừng

$DONG = DONG \cup \{n\}$ // đánh dấu n đã được xét

for các đỉnh kề v của n

if (v chưa đc xét) // v chưa ở trong DONG

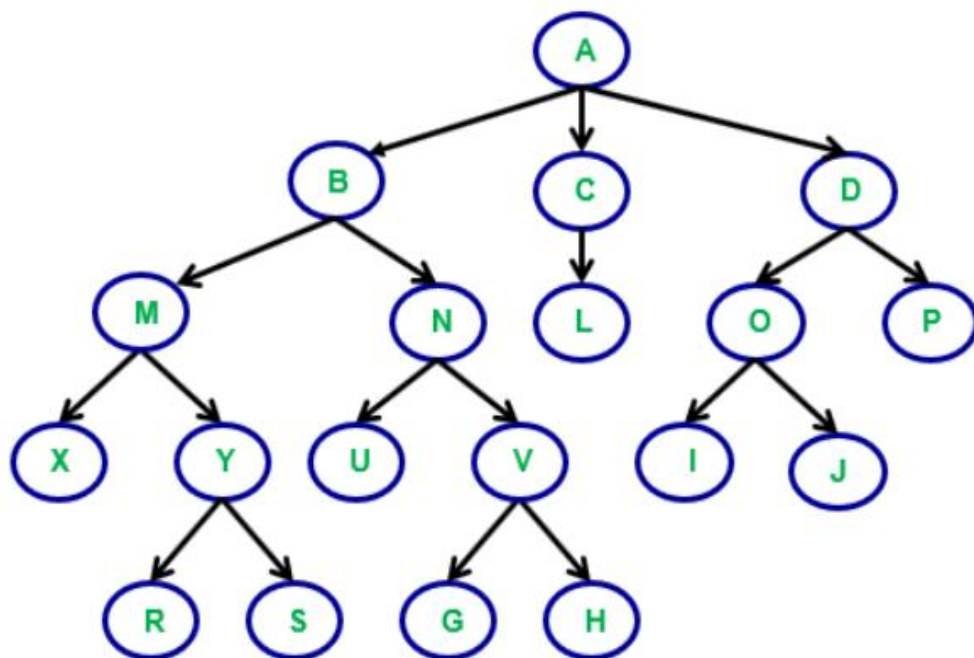
$MO = MO \cup \{v\}$ // đưa v vào đầu DS MO

$\text{father}(v) = n$ // lưu lại vết đường đi từ n đến v

}

1.2.1.3. Ví dụ thuật toán tìm kiếm theo chiều sâu

Cho đồ thị như hình vẽ sau:



Hình 1.1. Ví dụ thuật toán DFS

Đỉnh đầu $T_0 = A$, $T_G = \{R\}$

Tìm đường đi p từ T_0 đến T_G bằng phương pháp tìm kiếm theo chiều sâu?

Bảng 1.1. Ví dụ thuật giải DFS

n	B(n)	MO	DONG
---	------	----	------

		A	
A	B, C, D	B, C, D	A
B	M, N	M, N, C, D	A, B
M	X, Y	X, Y, N, C, D	A, B, M
X	Ø	Y, N, C, D	A, B, M, X
Y	R, S	R, S, N, C, D	A, B, M, X, Y
R	là đích -> dừng		

Xây dựng đường đi có hành trình: $p = A \rightarrow B \rightarrow M \rightarrow Y \rightarrow R$

Nhận xét:

- + Nếu trong đồ thị G tồn tại đường đi từ T_0 đến 1 đỉnh $T_G \in \text{Goal}$ thì hàm DFS sẽ dừng lại và cho đường đi p có độ dài có thể không ngắn nhất.
- + Với DFS các đỉnh được duyệt theo từng nhánh (theo chiều sâu).
- + Thuật toán DFS có độ phức tạp $O(b^d)$ với b là bậc của cây và d là chiều sâu của cây. Tuy nhiên trong trường hợp xấu nhất cũng là $O(b^d)$.

1.2.2. Thuật toán tìm kiếm theo chiều rộng (Breadth First Search)

1.2.2.1. Tư tưởng của chiến lược tìm kiếm theo chiều rộng

- Từ đỉnh xuất phát duyệt tất cả các đỉnh kề.
- Làm tương tự với các đỉnh vừa được duyệt.
- Quá trình duyệt kết thúc khi tìm thấy đỉnh T_G hoặc đã hết các đỉnh để duyệt.

1.2.2.2. Thuật toán tìm kiếm theo chiều rộng

Lưu trữ: Sử dụng hai danh sách DONG và MO hoạt động theo kiểu FIFO (hàng đợi).

DONG: Chứa các đỉnh đã xét

MO: chứa các đỉnh đang xét

1. $MO = \emptyset$; $MO = MO \cup \{T_0\}$

2. while ($MO \neq \emptyset$) {

$n = \text{get}(MO)$ // lấy đỉnh đầu trong danh sách MO

if ($n == T_G$) // nếu n là trạng thái kết thúc

return TRUE // tìm kiếm thành công, dừng

$DONG = DONG \cup \{n\}$ // đánh dấu n đã được xét

for các đỉnh kề v của n

if (v chưa đc xét) // v chưa ở trong DONG

$MO = MO \cup \{v\}$ // đưa v vào cuối DS MO

$$\text{father}(v) = n // \text{lưu lại vết đường đi từ } n \text{ đến } v \}$$

Nhận xét:

Trong tìm kiếm theo chiều rộng, trạng thái nào được sinh ra trước sẽ được phát triển trước, do đó danh sách MÔ được xử lý như hàng đợi. Trong bước 2, ta cần kiểm tra xem n có là trạng thái kết thúc hay không. Nói chung các trạng thái kết thúc được xác định bởi một số điều kiện nào đó, khi đó ta cần kiểm tra xem n có thỏa mãn các điều kiện đó hay không.

Nếu bài toán có nghiệm (tồn tại đường đi từ trạng thái ban đầu tới trạng thái đích), thì thuật toán tìm kiếm theo chiều rộng sẽ tìm ra nghiệm, đồng thời đường đi tìm được sẽ là ngắn nhất. Trong trường hợp bài toán vô nghiệm và không gian trạng thái hữu hạn, thuật toán sẽ dừng và cho thông báo vô nghiệm

Đánh giá tìm kiếm theo chiều rộng:

Bây giờ ta đánh giá thời gian và bộ nhớ mà tìm kiếm theo chiều rộng đòi hỏi. Giả sử, mỗi trạng thái khi được phát triển sẽ sinh ra b trạng thái kề. Ta sẽ gọi b là nhân tố nhánh. Giả sử rằng, nghiệm của bài toán là đường đi có độ dài d . Bởi nhiều nghiệm có thể được tìm ra tại một đỉnh bất kỳ ở mức d của cây tìm kiếm, do đó số đỉnh cần xem xét để tìm ra nghiệm là:

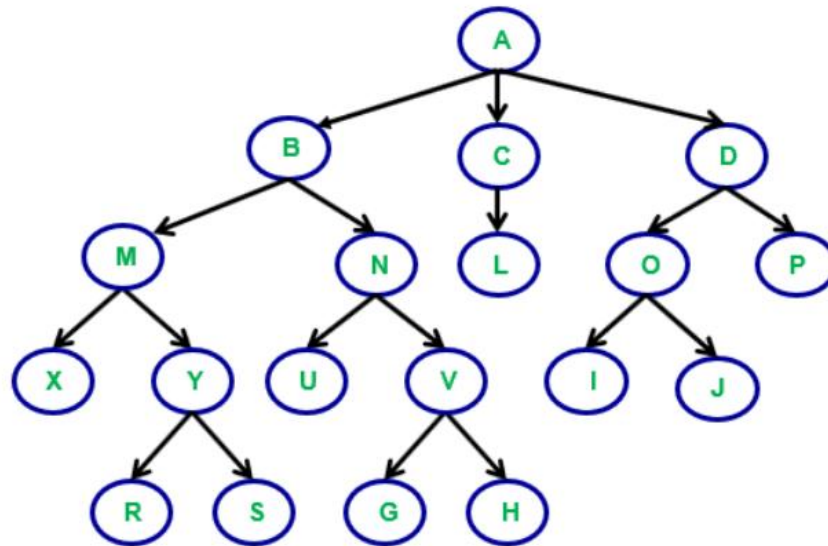
$$1 + b + b^2 + \dots + b^{d-1} + k$$

Trong đó k có thể là $1, 2, \dots, bd$. Do đó số lớn nhất các đỉnh cần xem xét là: $1 + b + b^2 + \dots + b^{d-1}$

Như vậy, độ phức tạp thời gian của thuật toán tìm kiếm theo chiều rộng là $O(b^d)$. Độ phức tạp không gian cũng là $O(b^d)$, bởi vì ta cần lưu vào danh sách MÔ tất cả các đỉnh của cây tìm kiếm ở mức d , số các đỉnh này là b^d .

1.2.2.3. Ví dụ thuật toán tìm kiếm theo chiều rộng

Cho đồ thị như hình vẽ sau:



Hình 1.2. Ví dụ thuật toán BFS

Đỉnh đầu $T_0=A$, $T_G= \{N\}$.

Tìm đường đi p từ T_0 đến T_G bằng phương pháp tìm kiếm theo chiều rộng?

Bảng 1. 2: Ví dụ thuật giải BFS

n	B(n)	MO	DONG
		A	
A	B, C, D	B, C, D	A
B	M, N	C, D, M, N	A, B
C	L	D, M, N, L	A, B, C
D	O, P	M, N, L, O, P	A, B, C, D
M	X, Y	N, L, O, P, X, Y	A, B, C, D, M
N	là đích -> dừng		

Xây dựng đường đi có hành trình: $p = A \rightarrow B \rightarrow N$.

Nhận xét:

- + Nếu trong đồ thị tồn tại đường đi từ T_0 đến 1 đỉnh $T_G \in \text{Goal}$ thì hàm BFS sẽ dừng lại và cho đường đi p có độ dài ngắn nhất.
- + Với BFS các đỉnh được duyệt theo từng mức (theo chiều rộng).
- + Thuật toán BFS có độ phức tạp $O(b^d)$ với b là bậc của cây và d là chiều sâu của cây.

1.3. Thuật toán Heuristic

- Khái niệm:

Trong tìm kiếm không gian trạng thái, heuristic là các luật dùng để chọn những nhánh nào có nhiều khả năng nhất dẫn đến một giải pháp chấp nhận

được Heuristic chỉ là một phỏng đoán chứa các thông tin về bước tiếp theo sẽ được chọn dùng trong việc giải quyết một vấn đề.

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 hoặc sai. Vì các heuristic sử dụng những thông tin hạn chế nên chúng ít khi có khả năng đoán trước chính xác cách hành xử của không gian trạng thái ở những giai đoạn xa hơn.

- *Chức năng của Heuristic:*

Các chương trình giải quyết những vấn đề trí tuệ nhân tạo sử dụng Heuristic cơ bản theo hai dạng:

- + Vấn đề có thể không có giải pháp chính xác vì những điều không rõ ràng trong diễn đạt vấn đề hoặc trong các dữ liệu có sẵn.
- + Vấn đề có thể có giải pháp chính xác, nhưng chi phí tính toán để tìm ra nó không cho phép.

- *Ưu điểm của Heuristic:*

Thuật giải Heuristic thể hiện cách giải bài toán với các đặc tính sau:

- + Thường tìm được lời giải tốt (Nhưng không chắc là lời giải tốt nhất).
- + Giải bài toán theo thuật giải Heuristic thường dễ dàng và nhanh chóng đưa ra kết quả hơn so với giải thuật tối ưu, vì vậy chi phí thấp hơn.
- + Thuật giải Heuristic thường thể hiện khá tự nhiên, gần gũi với cách suy nghĩ và hành động con người.

- *Phương pháp xây dựng thuật giải Heuristic:*

Thuật giải Heuristic gồm hai phần: Hàm đánh giá Heuristic và thuật toán để sử dụng nó trong tìm kiếm không gian trạng thái. Có nhiều các để xây dựng một thuật giải Heuristic, trong đó người ta thường dựa vào một số nguyên lý cơ bản như sau:

- + Nguyên lý vét cạn thông minh: Trong một bài toán tìm kiếm nào đó, khi không gian tìm kiếm lớn, ta thường tìm cách giới hạn lại không gian tìm kiếm hoặc thực hiện một kiểu dò tìm đặc biệt dựa vào đặc thù của bài toán để nhanh chóng tìm ra mục tiêu
- + Nguyên lý tham lam (Greedy): lấy tiêu chuẩn tối ưu (Trên phạm vi toàn cục) của bài toán để làm tiêu chuẩn chọn lựa hành động cho phạm vi cục bộ của từng bước (Hay từng giai đoạn) trong quá trình tìm kiếm lời giải.

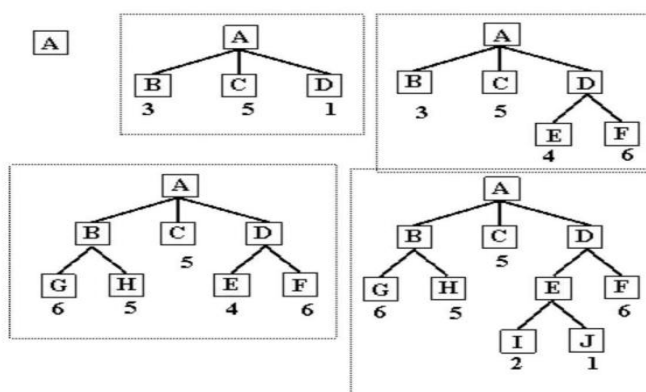
- + Nguyên lý thứ tự: thực hiện hành động dựa trên một cấu trúc thứ tự hợp lý của không gian khảo sát nhằm nhanh chóng đạt được một lời giải tốt.

1.3.1. Tìm kiếm tối ưu (Best-First-Search)

Ưu điểm của tìm kiếm theo chiều sâu là không phải quan tâm đến sự mở rộng của tất cả các nhánh. Ưu điểm của tìm kiếm chiều rộng là không bị sa vào các đường dẫn bế tắc (các nhánh cụt). Tìm kiếm tối ưu (Best-First Search) sẽ kết hợp hai phương pháp trên cho phép ta đi theo một con đường duy nhất tại một thời điểm, nhưng đồng thời vẫn xét được những hướng khác. Nếu con đường đang đi không triển vọng bằng những con đường đang quan sát, ta sẽ chuyển sang đi theo một trong số các con đường này.

Một cách cụ thể, tại mỗi bước của tìm kiếm BeFS, ta chọn đi theo trạng thái có khả năng cao nhất trong số các trạng thái đã được xét cho đến thời điểm đó. BeFS khác với tìm kiếm leo đồi là chỉ chọn trạng thái có khả năng cao nhất trong số các trạng thái kế tiếp có thể đến được t trạng thái hiện tại. Như vậy, với tiếp cận này, ta sẽ ưu tiên đi vào những nhánh tìm kiếm có khả năng nhất (giống tìm kiếm leo đồi), nhưng ta sẽ không bị luôn quan trọng các nhánh này vì nếu càng đi sâu vào một hướng mà ta phát hiện ra rằng hướng này càng đi thì càng xấu, đến mức nó xấu hơn cả những hướng mà ta chưa đi, thì ta sẽ không đi tiếp hướng hiện tại nữa mà chọn đi theo một hướng tốt nhất trong số những hướng chưa đi. Đó là tư tưởng chủ đạo của tìm kiếm tối ưu.

Ví dụ minh họa:



Hình 1.3. Ví dụ tìm kiếm tối ưu

Khởi đầu, chỉ có một nút (trạng thái) A nên nó sẽ được mở rộng tạo ra 3 nút mới B, C và D. Các con số dưới nút là giá trị cho biết độ tốt của nút. Con số càng nhỏ, nút càng tốt. Do D là nút có khả năng nhất nên nó sẽ được mở rộng tiếp sau nút A và sinh ra 2 nút kế tiếp là E và F. Đến đây, ta lại thấy nút B có vẻ có khả

năng nhất (trong các nút B,C,E,F) nên ta sẽ chọn mở rộng nút B và tạo ra 2 nút G và H. Nhưng lại một lần nữa, hai nút G, H này được đánh giá ít khả năng hơn E, vì thế sự chú ý lại trở về E. E được mở rộng và các nút được sinh ra từ E là I và J. Ở bước kế tiếp, J sẽ được mở rộng vì nó có khả năng nhất. Quá trình này tiếp tục cho đến khi tìm thấy một lời giải.

Để cài đặt các thuật giải theo kiểu tìm kiếm BeFS, thường cần dùng 2 tập hợp:

- OPEN: tập chứa các trạng thái đã được sinh ra nhưng chưa được xét đến (vì ta đã chọn một trạng thái khác). Thực ra, OPEN là một loại hàng đợi ưu tiên (priority queue) mà trong đó, phần tử có độ ưu tiên cao nhất là phần tử tốt nhất. Người ta thường cài đặt hàng đợi ưu tiên bằng Heap.
- CLOSE: tập chứa các trạng thái đã được xét đến. Chúng ta cần lưu trữ những trạng thái này trong bộ nhớ để đề phòng trường hợp khi một trạng thái mới được tạo ra lại trùng với một trạng thái mà ta đã xét đến trước đó. Trong trường hợp không gian tìm kiếm có dạng cây thì không cần dùng tập này.

Thuật giải

- Đặt OPEN chứa trạng thái khởi đầu.
- Cho đến khi tìm được trạng thái đích hoặc không còn nút nào trong OPEN, thực hiện:
 - + Chọn trạng thái tốt nhất (Tmax) trong OPEN (và xóa Tmax khỏi OPEN)
 - + Nếu Tmax là trạng thái kết thúc thì thoát.
 - + Ngược lại, tạo ra các trạng thái kế tiếp Tk có thể có từ trạng thái Tmax.

Đối với mỗi trạng thái kế tiếp Tk thực hiện:

Tính $f(T_k)$;

Thêm Tk vào OPEN

Nhận xét:

- BeFS khá đơn giản. Tuy vậy, trên thực tế, cũng như tìm kiếm chiều sâu và chiều rộng, hiếm khi ta dùng BeFS một cách trực tiếp.
- Thông thường, người ta thường dùng các phiên bản của BeFS là A^T , A^{KT} và A^* .

Thông tin về quá khứ và tương lai:

Thông thường, trong các phương án tìm kiếm theo kiểu BeFS, chi phí f của một trạng thái được tính dựa theo hai giá trị mà ta gọi là g và h . Trong đó h , như đã biết, đó là một ước lượng về chi phí từ trạng thái hiện hành cho đến trạng thái đích (thông tin tương lai), còn g là chiều dài quãng đường đã đi từ trạng thái ban đầu cho đến trạng thái hiện tại (thông tin quá khứ). Khi đó hàm ước lượng tổng chi phí $f(n)$ được tính theo công thức: $f(n) = g(n) + h(n)$

1.3.2. Thuật toán A^*

1.3.2.1. Khái niệm

Thuật giải A^* là một phương pháp tìm kiếm theo kiểu BeFS với chi phí của đỉnh là giá trị hàm g (tổng chiều dài thực sự của đường đi từ đỉnh bắt đầu đến đỉnh hiện tại).

Cho đồ thị $G = (V, E)$ với V : tập đỉnh; E : Tập cung. Với mỗi một cung người ta gán thêm một đại lượng được gọi là giá của cung.

$$C: E \rightarrow \mathbb{R}^+$$

$$e \rightarrow C(e)$$

Khi đó đường đi $p = n_1, n_2, \dots, n_k$ có giá được tính theo công thức:

$$Cost(p) = \sum_{i=1}^{k-1} Cost(n_i, n_{i+1})$$

Vấn đề đặt ra là tìm đường đi p từ T_0 đến đỉnh $T_G \in \text{Goal}$ sao cho $c(p) \rightarrow \min$

Vào: - Đồ thị $G = (V, E)$

$$C: E \rightarrow \mathbb{R}^+$$

$$e \rightarrow C(e)$$

- Đỉnh đầu T_0 và Goal chứa tập các đỉnh dọc

Ra: Đường đi $p: T_0 \rightarrow T_G \in \text{Goal}$ sao cho:

$$C(p) = g(n_k) = \min \{g(n)/n \in \text{Goal}\}.$$

1.3.2.2. Phương pháp: Sử dụng hai danh sách **CLOSE** và **OPEN**

```
void AT () {
```

```
    OPEN = {T0}, g(T0) = 0, CLOSE = ∅
```

```
    while (OPEN ≠ ∅) {
```

```
        n = getNew (OPEN) // Lấy đỉnh n sao cho g(n) đạt min
```

```
        if (n == TG) return True;
```

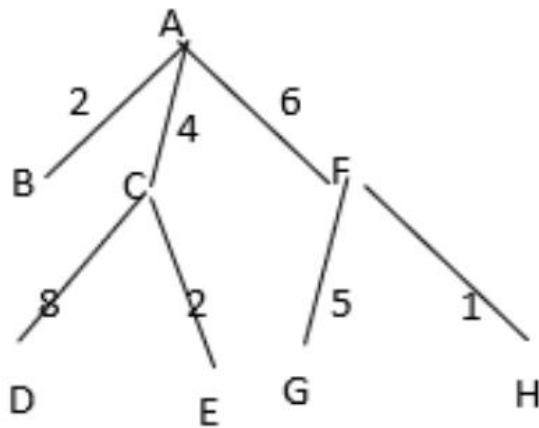
```
    else {
```

```

for (m ∈ B(n))
    if (m ∉ OPEN) && (m ∉ CLOSE) {
        g(m)=g(n) +cost (m, n);
        OPEN=OPEN ∪ {m};
    }
    else g(m) = min {g(m), gnew(m)};
CLOSE = CLOSE ∪ {n}
}
return False;
}

```

Ví dụ: Cho đồ thị. Đỉnh xuất phát A và Goal = {D, H}



Hình 1.4. Ví dụ thuật toán AT

Bảng 1.3. Ví dụ thuật giải AT

n	A (n)	OPEN	CLOSE
		A (0)	
A	B, C, F	B (2), C (4), F (6)	A
B	∅	C (4), F (6)	A, B
C	D, E	D (12), E (6), F (6)	A, B, C
E	∅	D (12), F (6)	A, B, C, E
F	G, H	G (11), H (7), D (12)	A, B, C, E, F
H	là đích -> dừng		

Trong bảng trên các con số trong ngoặc đơn là giá trị của hàm g (). Do $H \in \text{Goal}$ nên thuật toán dừng và đường đi tìm được p: $A \rightarrow F \rightarrow H$ có chi phí $C(p) = 7$.

Kết quả: Nếu trong đồ thị G tồn tại đường đi $p: T_0 \rightarrow T_G \in \text{Goal}$ thì thủ tục A^T sẽ dừng và cho kết quả đường đi có độ dài ngắn nhất.

Nhận xét:

- i) Nếu $C(a) = 1 \forall a \in E$ thì A^T trở thành BFS
- ii) Nếu thay điều kiện $g(n) \rightarrow \min$ bằng điều kiện $d(n) \rightarrow \max$ trong đó $d(n)$ là độ sâu hiện tại của đỉnh n . Khi đó A^T trở thành DFS.

1.3.3. Thuật toán AKT

Thuật giải A^T trong quá trình tìm đường đi chỉ xét đến các đỉnh và giá của chúng. Nghĩa là việc tìm đỉnh triển vọng chỉ phụ thuộc hàm $g(n)$ (thông tin quá khứ). Tuy nhiên thuật giải này không còn phù hợp khi gặp phải những bài toán phức tạp (độ phức tạp cấp hàm mũ) do ta phải tháo một lượng nút lớn. Để khắc phục nhược điểm này, người ta sử dụng thêm các thông tin bổ sung xuất phát từ bản thân bài toán để tìm ra các đỉnh có triển vọng, tức là đường đi tối ưu sẽ tập trung xung quanh đường đi tốt nhất nếu sử dụng các thông tin đặc tả về bài toán (thông tin quá tương lai).

Theo thuật giải này, chi phí của đỉnh được xác định:

$$f(n) = g(n) + h(n)$$

Đỉnh n được chọn nếu $f(n)$ đạt min.

Việc xác định hàm ước lượng $h(n)$ được thực hiện dựa theo:

- Chọn toán tử xây dựng cung sao cho có thể loại bớt các đỉnh không liên quan và tìm ra các đỉnh có triển vọng.
- Sử dụng thêm các thông tin bổ xung nhằm xây dựng tập OPEN và cách lấy các đỉnh trong tập OPEN.

Để làm được việc này người ta phải đưa ra độ đo, tiêu chuẩn để tìm ra các đỉnh có triển vọng. Các hàm sử dụng các kỹ thuật này gọi là hàm đánh giá. Sau đây là một số phương pháp xây dựng hàm đánh giá:

- Dựa vào xác suất của đỉnh trên đường đi tối ưu.
- Dựa vào khoảng cách, sự sai khác của trạng thái đang xét với trạng thái đích hoặc các thông tin liên quan đến trạng thái đích.

1.3.3.1. Thuật toán

- Vào:
- Đồ thị $G = (V, E)$ trong đó V là tập đỉnh, E là tập cung.
 - $f: V \rightarrow R^+$ ($f(n)$: hàm ước lượng)
 - Đỉnh đầu T_0 và tập các đỉnh đích

Ra: - Đường đi p: $T_0 \rightarrow T_G \in \text{Goal}$

Phương pháp: Sử dụng 2 danh sách CLOSE và OPEN

```
void AKT () {  
    OPEN = {T0},  
    g(T0) = 0  
    Tính h(T0), f(T0) = g(T0) + h(T0)  
    while (OPEN !=  $\emptyset$ ) {  
        n = getNew (OPEN) // lấy đỉnh n sao cho f(n) đạt min  
        if (n == TG) return True;  
        else {  
            for (m  $\in$  B(n)) {  
                g(m) = g(n) + cost (m, n);  
                Tính h(m), f(m)=g(m)+h(m);  
                OPEN = OPEN  $\cup$  {m};  
            }  
        }  
    }  
    return False;  
}
```

1.3.3.2. Ví dụ

Xét bài toán tháp Hà Nội với $n = 2$

Chọn hàm $f(n) = g(n) + h(n)$

$g(n)$ là số lần chuyển đĩa từ T_0 đến trạng thái n.

$h(n)$ là thông tin liên quan đến số đĩa ở cọc 3.

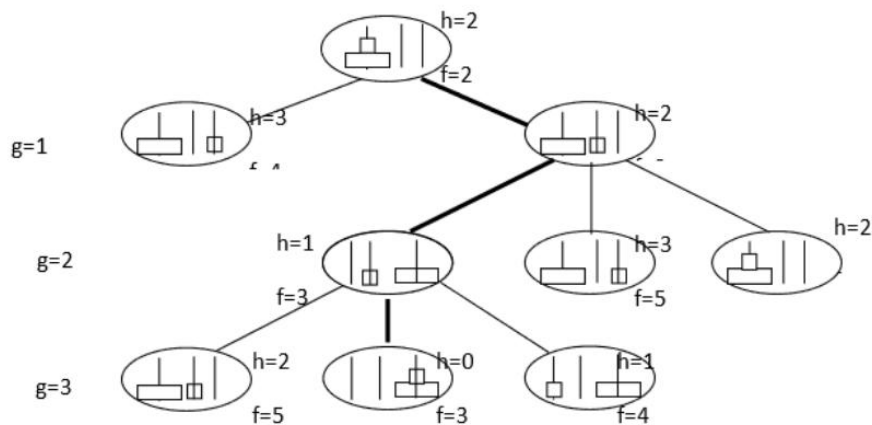
Giá trị $h(n)$ ở các trạng thái như sau:

Nếu cọc 3 chưa có đĩa nào thì $h = 2$

Nếu cọc 3 có 1 đĩa nhỏ thì $h = 3$

Nếu cọc 3 có 1 đĩa to thì $h = 1$

Nếu cọc 3 có 2 đĩa và đĩa nhỏ ở trên đĩa to thì $h = 0$



Hình 1.5. Ví dụ thuật giải AKT

1.3.4. Thuật giải A^*

1.3.4.1 Khái niệm

A^* là một phiên bản đặc biệt của A^{KT} áp dụng cho trường hợp đồ thị. Thuật toán này tìm một đường đi từ một nút khởi đầu tới một nút đích cho trước (hoặc tới một nút thỏa mãn một điều kiện đích). Thuật toán này sử dụng một "đánh giá heuristic" để xếp loại từng nút theo ước lượng về tuyến đường tốt nhất đi qua nút đó. Thuật toán này duyệt các nút theo thứ tự của đánh giá heuristic này.

Ý tưởng trực quan: Xét bài toán tìm đường - bài toán mà A^* thường được dùng để giải. A^* xây dựng tăng dần tất cả các tuyến đường từ điểm xuất phát cho tới khi nó tìm thấy một đường đi chạm tới đích. Tuy nhiên, cũng như tất cả các thuật toán tìm kiếm có thông tin (informed tìm kiếm thuật toán), nó chỉ xây dựng các tuyến đường "có vẻ" dẫn về phía đích.

Để biết những tuyến đường nào có khả năng sẽ dẫn tới đích, A^* sử dụng một "đánh giá heuristic" về khoảng cách từ điểm bất kỳ cho trước tới đích. Trong trường hợp tìm đường đi, đánh giá này có thể là khoảng cách đường chim bay - một đánh giá xấp xỉ thường dùng cho khoảng cách của đường giao thông.

Điểm khác biệt của A^* đối với tìm kiếm theo lựa chọn tốt nhất là nó còn tính đến khoảng cách đã đi qua. Điều đó làm cho A^* "đầy đủ" và "tối ưu", nghĩa là, A^* sẽ luôn luôn tìm thấy đường đi ngắn nhất nếu tồn tại một đường đi như vậy. A^* không đảm bảo sẽ chạy nhanh hơn các thuật toán tìm kiếm đơn giản hơn. Trong một môi trường dạng mê cung, cách duy nhất để

đến đích có thể là trước hết phải đi về phía xa đích và cuối cùng mới quay lại. Trong trường hợp đó, việc thử các nút theo thứ tự "gần đích hơn thì được thử trước" có thể gây tốn thời gian.

Thuật toán A*: A* lưu giữ một tập các lời giải chưa hoàn chỉnh, nghĩa là các đường đi qua đồ thị, bắt đầu từ nút xuất phát. Tập lời giải này được lưu trong một hàng đợi ưu tiên (priority queue). Thứ tự ưu tiên gán cho một đường đi được quyết định bởi hàm.

Trong đó, là chi phí của đường đi cho đến thời điểm hiện tại, nghĩa là tổng trọng số của các cạnh đã đi qua. là hàm đánh giá heuristic về chi phí nhỏ nhất để đến đích từ. Ví dụ, nếu "chi phí" được tính là khoảng cách đã đi qua, khoảng cách đường chim bay giữa hai điểm trên một bản đồ là một đánh giá heuristic cho khoảng cách còn phải đi tiếp.

```
void Astar () {
    OPEN = {T0}, CLOSE =  $\emptyset$ , g(T0) = 0,
    Tính h(T0), f(T0) = g(T0) + h(T0);
    while (OPEN !=  $\emptyset$ ) {
        n = getnew (OPEN) // lấy đỉnh n sao cho f(n) đạt min.
        if (n == TG) return True;
        else {
            for (m ∈ B(n))
                if (m ∈ (OPEN ∪ CLOSE)) {
                    Tính h(m), g(m), f(m) = g(m) + h(m);
                    Cha(m) = n; OPEN = OPEN ∪ {m};
                }
            else {
                g (m) = min {gold(m), gnew(m)};
                Cập nhật lại OPEN;
            }
        }
        CLOSE = CLOSE ∪ {n};
    }
    return false;}

```

1.3.4.2. Các tính chất

Cũng như tìm kiếm theo chiều rộng (breadth-first search), A* là thuật toán đầy đủ (complete) theo nghĩa rằng nó sẽ luôn luôn tìm thấy một lời giải nếu bài toán có lời giải.

Nếu hàm heuristic h có tính chất thu nạp được (admissible), nghĩa là nó không bao giờ đánh giá cao hơn chi phí nhỏ nhất thực sự của việc đi tới đích, thì bản thân A* có tính chất thu nạp được (hay tối ưu) nếu sử dụng một tập đóng. Nếu không sử dụng tập đóng thì hàm h phải có tính chất đơn điệu (hay nhất quán) thì A* mới có tính chất tối ưu. Nghĩa là nó không bao giờ đánh giá chi phí đi từ một nút tới một nút kế nó cao hơn chi phí thực. Phát biểu một cách hình thức, với mọi nút x, y trong đó y là nút tiếp theo của x :

$$h(x) \leq g(y) - g(x) + h(y)$$

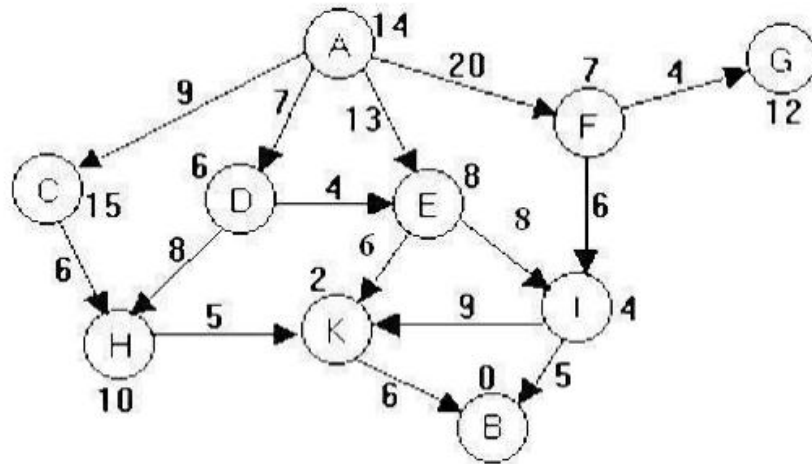
A* còn có tính chất hiệu quả một cách tối ưu (optimally efficient) với mọi hàm heuristic, có nghĩa là không có thuật toán nào cũng sử dụng hàm heuristic đó mà chỉ phải mở rộng ít nút hơn A*, trừ khi có một số lời giải chưa đầy đủ mà tại đó dự đoán chính xác chi phí của đường đi tối ưu.

Ưu điểm Một thuật giải linh động, tổng quát, trong đó hàm chứa cả tìm kiếm chiều sâu, tìm kiếm chiều rộng và những nguyên lý Heuristic khác. Nhanh chóng tìm đến lời giải với sự định hướng của hàm Heuristic. Chính vì thế mà người ta thường nói A* chính là thuật giải tiêu biểu cho Heuristic.

Nhược điểm A* rất linh động nhưng vẫn gặp một khuyết điểm cơ bản - giống như chiến lược tìm kiếm chiều rộng - đó là tốn khá nhiều bộ nhớ để lưu lại những trạng thái đã đi qua.

1.3.4.3. Ví dụ

Trạng thái ban đầu $T_0 = A$, trạng thái đích $Goal = \{B\}$, các số ghi cạnh các cung là độ dài đường đi, các số cạnh các đỉnh là giá trị của hàm h



Hình 1.6. Ví dụ thuật giải A^*

Ban đầu $OPEN = \{A, g(A) = 0, f(A) = 14\}$

Phát triển đỉnh A sinh ra các đỉnh con C, D, E và F. Tính giá trị của hàm f tại các đỉnh này ta có:

$$\begin{aligned}
 OPEN = \{ \\
 &g(C) = 9, f(C) = 9 + 15 = 24, \text{cha}(C) = A, g(D) = 7, \\
 &f(D) = 7 + 6 = 13, \text{cha}(D) = A, \\
 &g(E) = 13, f(E) = 13 + 8 = 21, \text{cha}(E) = A, g(F) = 20, \\
 &f(F) = 20 + 7 = 27, \text{cha}(F) = A \\
 &\} \\
 CLOSE = \{A, g(A) = 0, f(A) = 14\}
 \end{aligned}$$

Do $f(D) = 13$ nhỏ nhất nên chọn D để phát triển. Phát triển D, ta nhận được các đỉnh kế tiếp H và E.

$g(H) = g(D) + \text{cost}(D, H) = 7 + 8 = 15, f(H) = 15 + 10 = 25. g(E) = g(D) + \text{cost}(D, E) = 7 + 4 = 11, f(E) = 11 + 8 = 19.$

Bây giờ ta bổ sung hai đỉnh mới này vào tập OPEN. Tuy nhiên trong tập OPEN lúc này đã có đỉnh E nên cần phải so sánh $g(E)$ đã có và $g(E)$ vừa tính được. Ta so sánh hai giá trị này và giữ lại giá trị nhỏ hơn. Do vậy

$$\begin{aligned}
 OPEN = \{ \\
 &g(C) = 9, f(C) = 9 + 15 = 24, \text{cha}(C) = A, \\
 &g(E) = 11, f(E) = 11 + 8 = 19, \text{cha}(E) = D, \\
 &g(F) = 20, f(F) = 20 + 7 = 27, \text{cha}(F) = A, \\
 &g(H) = 15, f(H) = 15 + 10 = 25, \text{cha}(H) = D \\
 &\}
 \end{aligned}$$

$CLOSE = \{A, g(A) = 0, f(A) = 14,$

$$D, g(D) = 7, f(D) = 7 + 6 = 13, \text{cha}(D) = A$$

}

Với lập luận tương tự như trên ta chọn đỉnh E để phát triển. Các đỉnh kế tiếp của E là K và I:

$$\text{OPEN} = \{$$

$$g(C) = 9, f(C) = 9 + 15 = 24, \text{cha}(C) = A,$$

$$g(F) = 20, f(F) = 20 + 7 = 27, \text{cha}(F) = A,$$

$$g(H) = 15, f(H) = 15 + 10 = 25, \text{cha}(H) = D,$$

$$g(K) = 17, f(K) = 17 + 2 = 19, \text{cha}(K) = E,$$

$$g(I) = 19, f(I) = 19 + 4 = 23, \text{cha}(I) = E$$

}

$$\text{CLOSE} = \{A, g(A) = 0, f(A) = 14,$$

$$D, g(D) = 7, f(D) = 7 + 6 = 13, \text{cha}(D) = A,$$

$$E, g(E) = 11, f(E) = 11 + 8 = 19, \text{cha}(E) = D$$

}

Chọn đỉnh K để phát triển. Các đỉnh tiếp của K là B.

$$\text{OPEN} = \{$$

$$g(C) = 9, f(C) = 9 + 15 = 24, \text{cha}(C) = A,$$

$$g(F) = 20, f(F) = 20 + 7 = 27, \text{cha}(F) = A,$$

$$g(H) = 15, f(H) = 15 + 10 = 25, \text{cha}(H) = D,$$

$$g(I) = 19, f(I) = 19 + 2 = 23, \text{cha}(I) = E,$$

$$g(B) = 23, f(B) = 23 + 0 = 23, \text{cha}(B) = K$$

}

$$\text{CLOSE} = \{A, g(A) = 0, f(A) = 14,$$

$$D, g(D) = 7, f(D) = 7 + 6 = 13, \text{cha}(D) = A,$$

$$E, g(E) = 11, f(E) = 11 + 8 = 19, \text{cha}(E) = D,$$

$$K, g(K) = 17, f(K) = 17 + 2 = 19, \text{cha}(K) = E$$

}

Trong tập OPEN có $f(B) = f(I)$ nên chọn ngẫu nhiên một trong hai đỉnh này. giả sử chọn đỉnh B để phát triển.

Do $B \in \text{Goal}$ nên quá trình tìm kiếm kết thúc. Để đưa ra đường đi ta thử ngược lại trong tập CLOSE. Khi đó đường đi tìm được có chi phí $c(p) = 23$ và trình tự các đỉnh.

$$p: A \rightarrow D \rightarrow E \rightarrow K \rightarrow B$$

Nhận xét:

- i) Đường đi tìm được có thể không phải là tốt nhất.
- ii) Nếu $h(n) = 0$ trong mọi trường hợp thì A^* trở thành A^T .

Bảng so sánh 2 thuật toán DFS và BFS:

Bảng 1.4. So sánh thuật toán DFS và BFS

	BFS	DFS
Thứ tự các đỉnh khi duyệt đồ thị	Các đỉnh được duyệt theo từng mức	Các đỉnh được duyệt theo từng nhánh
Độ dài đường đi p từ T_0 đến T_G	Ngắn nhất	Có thể không ngắn nhất
Tính hiệu quả	<ul style="list-style-type: none"> - Chiến lược có hiệu quả khi lời giải nằm ở mức thấp (gần gốc cây) - Thuận lợi khi tìm kiếm nhiều lời giải 	<ul style="list-style-type: none"> - Chiến lược có hiệu quả khi lời giải nằm gần hướng đi được chọn theo phương án - Thuận lợi khi tìm kiếm 1 lời giải
Sử dụng bộ nhớ	Lưu trữ toàn bộ KGTT	Lưu trữ các TT đang xét
Trường hợp tốt nhất	Vết cạn toàn bộ	Phương án chọn đường đi chính xác có lời giải trực tiếp
Trường hợp xấu nhất	Vết cạn	Vết cạn

CHƯƠNG 2. XÂY DỰNG CHƯƠNG TRÌNH

2.1. Giới thiệu bài toán tic tac toe

Tic Tac Toe, còn được gọi là “xoăn ốc” hoặc “mô hình”, là trò chơi giải đố vui nhỏ thường được chơi trên một bảng vuông 3×3 . Trò chơi này thường được hai người chơi lượt đi lượt lại, mỗi người đại diện cho một biểu tượng khác nhau – thường là “X” và “O”. Mục tiêu của trò chơi là xếp thành công ba biểu tượng của mình theo hàng ngang, hàng dọc hoặc đường chéo trên bảng, trước khi bảng đầy hoặc không thể thực hiện bước đi tiếp theo.

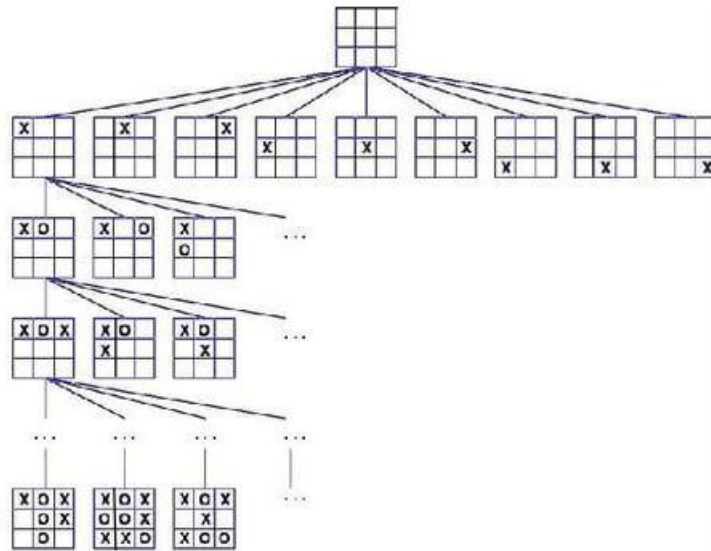
- Luật chơi:
- + Trò chơi diễn ra trên một bảng 3×3 .
- + Hai người chơi thay phiên nhau thực hiện lượt đi. Mỗi lượt đi, người chơi sẽ đặt biểu tượng của mình vào một ô trống trên bảng. Mục tiêu là tạo thành một chuỗi gồm ba biểu tượng của mình theo hàng ngang, hàng dọc hoặc đường chéo.
- + Trò chơi kết thúc khi một trong hai người chơi đã xếp thành công ba biểu tượng của mình theo hàng ngang, hàng dọc hoặc đường chéo trên bảng. Điều này có nghĩa là nếu ba biểu tượng của một người chơi xuất hiện liên tiếp trên một hàng ngang, một hàng dọc hoặc đường chéo của bảng, thì người chơi đó sẽ giành chiến thắng. Nếu cả hai người chơi đều không tạo thành chuỗi ba biểu tượng liên tiếp và bảng đã đầy, trò chơi sẽ kết thúc là hòa.

2.2. Mô hình hóa bài toán

2.2.1. Không gian trạng thái

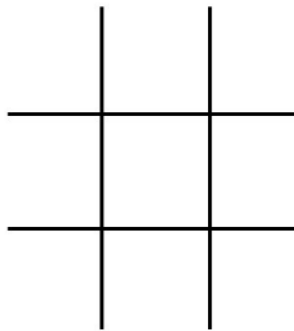
Mỗi trạng thái được biểu diễn bằng bảng 3×3 , trong đó mỗi ô có thể là:

- Ký hiệu “X”: ô đã được đánh dấu bởi người chơi X
- Ký hiệu “O”: ô đã được đánh dấu bởi người chơi O
- Trống: ô chưa được đánh dấu



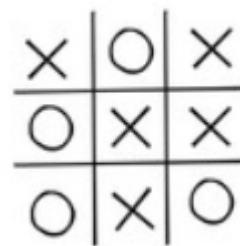
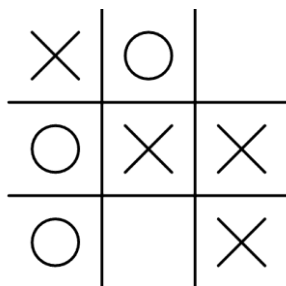
Hình 2.1. Mô tả không gian trạng thái bài toán tic tac toe

+ Trạng thái đầu: Bảng trống.



Hình 2.2. Trạng thái đầu của bài toán

+ Trạng thái đích: Trạng thái mà một trong hai người chơi đã thắng hoặc bảng đầy ô mà không có người thắng (hòa).



Hình 2.3. Trạng thái đích của bài toán

2.2.2. Toán tử chuyển trạng thái

Trong trò chơi Tic Tac Toe, mỗi hành động của người chơi (đánh một ký hiệu vào ô trống) được coi là một toán tử chuyển trạng thái, đưa bàn cờ từ trạng thái hiện tại sang trạng thái mới.

Một hành động hợp lệ phải thỏa mãn các điều kiện sau:

- Phạm vi hợp lệ: Vị trí đặt dấu phải nằm trong phạm vi bàn cờ 3x3: Hành động chỉ được thực hiện tại các ô có tọa độ từ (0,0) đến (2,2).
- Ô được chọn phải còn trống: Người chơi chỉ được đặt dấu (X hoặc O) vào ô chưa có dấu nào.
- Luân phiên lượt chơi: Người chơi X và O đánh luân phiên nhau, không ai được đánh hai lần liên tiếp.
- Trò chơi kết thúc: Nếu đã có người thắng (ba dấu thẳng hàng) hoặc bảng đã đầy (hòa), thì không còn hành động nào được thực hiện nữa.

Ví dụ:

1) Trạng thái cơ bản:

- Lựa chọn người chơi đầu là X:

+ Hành động hợp lệ:

- Đặt X vào ô (0,0)
- Đặt X vào ô (0,1)
- Đặt X vào ô (0,2)
- Đặt X vào ô (1,0)
- Đặt X vào ô (1,1)
- Đặt X vào ô (1,2)
- Đặt X vào ô (2,0)
- Đặt X vào ô (2,1)
- Đặt X vào ô (2,2)

Giả sử X đánh vào ô (0,0)

2) Trạng thái bàn cờ hiện tại:

X		

- Người chơi tiếp theo: O

+ Hành động hợp lệ:

- Đặt O vào ô (0,1)
- Đặt O vào ô (0,2)
- Đặt O vào ô (1,0)
- Đặt O vào ô (1,1)
- Đặt O vào ô (1,2)
- Đặt O vào ô (2,0)
- Đặt O vào ô (2,1)
- Đặt O vào ô (2,2)

Giả sử O đánh vào ô (0,1)

3) Trạng thái bàn cờ hiện tại:

X	O	

- Người chơi tiếp theo: X

+ Hành động hợp lệ:

- Đặt X vào ô (0,2)
- Đặt X vào ô (1,0)
- Đặt X vào ô (1,1)
- Đặt X vào ô (1,2)
- Đặt X vào ô (2,0)
- Đặt X vào ô (2,1)
- Đặt X vào ô (2,2)

Giả sử X đánh vào ô (0,2)

4) Trạng thái bàn cờ hiện tại:

X	O	X

- Người chơi tiếp theo: O

+ Hành động hợp lệ:

- Đặt O vào ô (1,0)

- Đặt O vào ô (1,1)
- Đặt O vào ô (1,2)
- Đặt O vào ô (2,0)
- Đặt O vào ô (2,1)
- Đặt O vào ô (2,2)

Giả sử O đánh vào ô (1,2)

5) Trạng thái bàn cờ hiện tại:

X	O	X
	O	

- Người chơi tiếp theo: X

+ Hành động hợp lệ:

- Đặt X vào ô (1,0)
- Đặt X vào ô (1,2)
- Đặt X vào ô (2,0)
- Đặt X vào ô (2,1)
- Đặt X vào ô (2,2)

Giả sử X đánh vào ô (1,2)

6) Trạng thái bàn cờ hiện tại:

X	O	X
	O	X

- Người chơi tiếp theo: O

+ Hành động hợp lệ:

- Đặt O vào ô (1,0)
- Đặt O vào ô (2,0)
- Đặt O vào ô (2,1)
- Đặt O vào ô (2,2)

Giả sử O đánh vào ô (2,1)

7) Trạng thái bàn cờ kết thúc:

X	O	X
	O	X
	O	

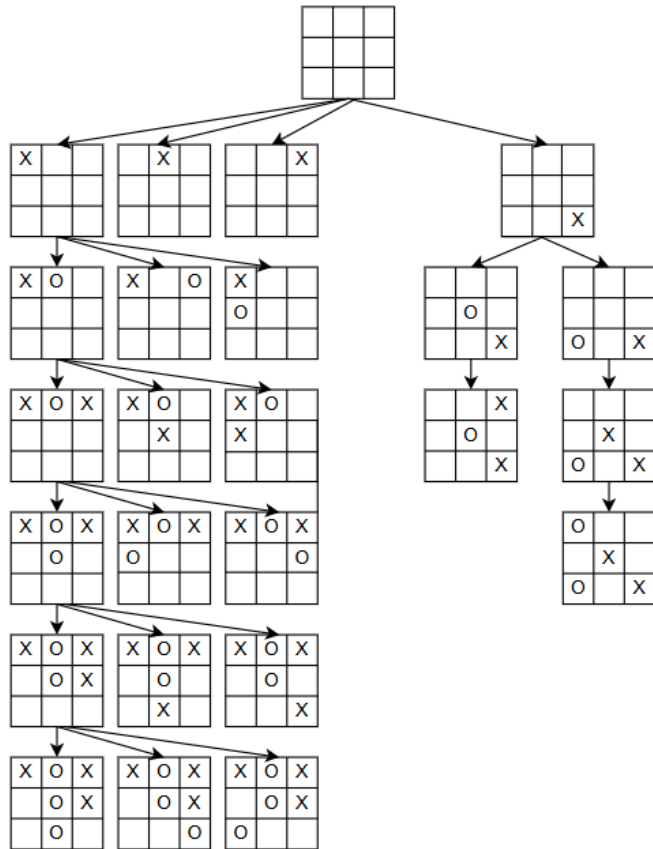
- Người chơi O chiến thắng

2.3. Cài đặt thuật toán DFS cho tic tac toe

2.3.1. Ý tưởng sử dụng DFS

- AI sẽ duyệt cây trạng thái theo chiều sâu, giả định tất cả các nước đi có thể xảy ra.
- Mỗi trạng thái con được sinh từ trạng thái cha thông qua một nước đi hợp lệ.
- Khi đến trạng thái đích, đánh giá kết quả (thắng, thua, hòa).
- DFS sẽ quay lui và đánh giá giá trị tốt nhất để chọn nước đi hiện tại.

Mình họa cây trạng thái sinh ra từ trạng thái ban đầu (bảng trống), AI (người chơi X) thực hiện lần lượt các nước đi hợp lệ. Mỗi nhánh đại diện cho một hành động chuyển trạng thái. Cây được mở rộng theo chiều sâu – mỗi nhánh sẽ đi sâu đến trạng thái kết thúc trước khi quay lui để duyệt nhánh khác. DFS giúp đánh giá toàn bộ các khả năng thắng, thua, hòa và chọn đường đi tối ưu nhất.



Hình 2.4. Cây trạng thái trò chơi Tic Tac Toe sử dụng thuật toán DFS

2.3.2. Cài đặt thuật toán

- Hiển thị bàn cờ

```
def in_board(board):
    for hang in board:
        print("|".join(hang))
    print("-" * 5)
```

Hình 2.5. Thuật toán hiển thị bàn cờ

- Kiểm tra thắng

```
def kiem_tra_thang(board, nguoi_choi):
    dieu_kien_thang = [
        [(0,0), (0,1), (0,2)],
        [(1,0), (1,1), (1,2)],
        [(2,0), (2,1), (2,2)],
        [(0,0), (1,0), (2,0)],
        [(0,1), (1,1), (2,1)],
        [(0,2), (1,2), (2,2)],
        [(0,0), (1,1), (2,2)],
        [(0,2), (1,1), (2,0)],
    ]
    for dieu_kien in dieu_kien_thang:
        if all(board[h][c] == nguoi_choi for h, c in dieu_kien):
            return True
    return False
```

Hình 2.6. Kiểm tra chiến thắng

- Kiểm tra hòa

```
def kiem_tra_hoa(board):
    return all(o != ' ' for hang in board for o in hang)
```

Hình 2.7. Kiểm tra hòa cờ

- Hàm DFS – tìm nước đi tối ưu

```

def tim_kiem_dfs(board, nguoi_choi):
    if kiem_tra_thang(board, 'X'):
        return 1
    if kiem_tra_thang(board, 'O'):
        return -1
    if kiem_tra_hoa(board):
        return 0
    nguoi_tiep_theo = 'O' if nguoi_choi == 'X' else 'X'
    cac_diem = []
    for hang in range(3):
        for cot in range(3):
            if board[hang][cot] == ' ':
                board[hang][cot] = nguoi_choi
                diem = tim_kiem_dfs(board, nguoi_tiep_theo)
                cac_diem.append(diem)
                board[hang][cot] = ' '
    if not cac_diem:
        return 0
    return max(cac_diem) if nguoi_choi == 'X' else min(cac_diem)

```

Hình 2.8. Tìm nước đi tối ưu

- Tìm nước đi tốt nhất cho AI (X)

```

def AI_move(board):
    diem_tot_nhat = -float('inf')
    move = None

    for hang in range(3):
        for cot in range(3):
            if board[hang][cot] == ' ':
                board[hang][cot] = 'X'
                diem = tim_kiem_dfs(board, 'O')
                board[hang][cot] = ' '
                if diem > diem_tot_nhat:
                    diem_tot_nhat = diem
                    move = (hang, cot)

    if move:
        board[move[0]][move[1]] = 'X'

```


Hình 2.9. AI chọn nước đi tối ưu

- Cho phép người chơi nhập nước đi

```
def nguoi_choi_move(board):  
    while True:  
        try:  
            vi_tri = input("Nhập vị trí đánh (hàng cột), mỗi giá trị từ 0 đến 2, cách nhau bằng dấu cách: ")  
            hang, cot = map(int, vi_tri.strip().split())  
  
            if 0 <= hang < 3 and 0 <= cot < 3 and board[hang][cot] == ' ':  
                board[hang][cot] = 'O'  
                break  
            else:  
                print(" Vị trí không hợp lệ hoặc ô đã có người đánh. Vui lòng thử lại.")  
        except:  
            print(" Lỗi định dạng! Nhập lại theo cú pháp: số_hàng số_cột (VD: 1 2)")
```

Hình 2.10. Xử lý nước đi người chơi

- Hàm main()

```

def main():
    board = [[' ']*3 for _ in range(3)]

    print(" Bạn là người chơi 'O', AI là 'X'.")
    in_board(board)

    while True:
        nguoi_choi_move(board)
        print("\n Bàn cờ sau lượt đi của bạn:")
        in_board(board)

        if kiem_tra_thang(board, 'O'):
            print(" Bạn đã chiến thắng!")
            break

        if kiem_tra_hoa(board):
            print("Trò chơi hòa. ")
            break

        AI_move(board)
        print(" Bàn cờ sau lượt đi của AI:")
        in_board(board)

        if kiem_tra_thang(board, 'X'):
            print(" AI đã thắng!")
            break

        if kiem_tra_hoa(board):
            print(" Trò chơi hòa.")
            break

if __name__ == "__main__":
    main()

```

Hình 2.11. Hàm main

2.4. Kết quả

```
Bạn chơi O, AI chơi X.
| |
----
| |
----
| |
----
Nhập vị trí đánh (hàng,cột) từ 0 đến 2, cách nhau bằng dấu cách: 0 0
Bàn cờ sau nước của bạn:
O| |
----
| |
----
| |
----
AI đang tính nước đi...
Bàn cờ sau nước của AI:
O| |
----
|X|
----
| |
-----
```

Hình 2.12. Kết quả chạy thử

```
Nhập vị trí đánh (hàng,cột) từ 0 đến 2, cách nhau bằng dấu cách: 0 1
Bàn cờ sau nước của bạn:
O|O|
----
|X|
----
| |
----
AI đang tính nước đi...
Bàn cờ sau nước của AI:
O|O|X
----
|X|
----
| |
----
Nhập vị trí đánh (hàng,cột) từ 0 đến 2, cách nhau bằng dấu cách: 1 0
Bàn cờ sau nước của bạn:
O|O|X
----
O|X|
----
| |
-----
```

Hình 2.13. Kết quả chạy thử

```
AI đang tính nước đi...  
Bàn cờ sau nước của AI:  
o|o|x  
----  
o|x|  
----  
x| |  
----  
AI thắng! 🤔
```

Hình 2.14. Kết quả chạy thử

KẾT LUẬN

Qua quá trình tìm hiểu và thực hiện đề tài “Tìm hiểu thuật toán tìm kiếm mù và ứng dụng vào bài toán Tic Tac Toe”, nhóm chúng em đã có cơ hội nghiên cứu sâu hơn về các phương pháp tìm kiếm trong không gian trạng thái, đặc biệt là các thuật toán tìm kiếm mù như DFS, BFS và các thuật toán Heuristic như AT, AKT, A*. Việc phân tích chi tiết đặc điểm, ưu - nhược điểm và cách cài đặt của từng thuật toán giúp nhóm hiểu rõ hơn về cơ chế hoạt động và ứng dụng của chúng trong thực tế.

Bên cạnh phần lý thuyết, nhóm cũng đã xây dựng thành công chương trình mô phỏng trò chơi Tic Tac Toe có sử dụng thuật toán tìm kiếm DFS. Kết quả thu được minh họa rõ ràng cách mà trí tuệ nhân tạo có thể áp dụng để đưa ra các nước đi tối ưu, từ đó góp phần thể hiện tính ứng dụng thực tiễn của AI trong các bài toán đơn giản nhưng đầy ý nghĩa.

Mặc dù đã cố gắng hoàn thiện bài báo cáo một cách nghiêm túc và khoa học, nhóm nhận thấy vẫn còn một số hạn chế nhất định do kinh nghiệm và thời gian thực hiện còn hạn chế. Rất mong nhận được sự góp ý của cô để nhóm có thể rút ra thêm nhiều kinh nghiệm cho những lần thực hiện sau.

Nhóm em xin chân thành cảm ơn!

TÀI LIỆU THAM KHẢO

- [1] Nguyễn Phương Nga, Trần Hùng Cường, *Giáo trình Trí tuệ nhân tạo*, NXB Thống kê, 2021.
- [2] Nguyễn Thanh Thủy, *Trí tuệ nhân tạo*, NXB Thống kê, 1999.
- [3] Nguyễn Đình Thức, *Giáo trình Trí tuệ nhân tạo: Mạng nơ-ron phương pháp và ứng dụng*, NXB Giáo dục, 2000.
- 1) <https://luatduonggia.vn/tic-tac-toe-la-gi-cach-choi-tic-tac-toe-lam-sao-de-thang/>

KẾ HOẠCH THỰC HIỆN TIỂU LUẬN, BÀI TẬP LỚN, ĐỒ ÁN/DỰ ÁN

Tên lớp: 20242IT6094003

Khoá: 18

Họ và tên sinh viên: Trần Đào Duy Anh

Tên chủ đề: Tìm hiểu các thuật toán tìm kiếm mù và ứng dụng vào bài toán tic tac toe

Tuần	Người thực hiện	Nội dung công việc	Phương pháp thực hiện
1	Trần Đào Duy Anh	Trình bày về thuật toán tìm kiếm mù	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
2	Trần Đào Duy Anh	Trình bày về tìm kiếm tối ưu	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
3	Trần Đào Duy Anh	Mô tả không gian trạng thái của bài toán được chọn và toán tử chuyển trạng thái	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
4	Trần Đào Duy Anh	Tham gia khởi tạo các biến cần thiết và xây dựng thuật toán tìm kiếm DFS	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
5	Trần Đào Duy Anh	Lập trình để giải quyết bài toán đề ra	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
6	Trần Đào Duy Anh	Góp ý kiến, sửa đổi để hoàn thiện báo cáo và các biểu mẫu, phiếu liên quan	Tổng hợp tất cả các nội dung báo cáo, các biểu mẫu, phiếu liên quan

Ngày tháng năm 2025

XÁC NHẬN CỦA GIẢNG VIÊN

(Ký, ghi rõ họ tên)

KẾ HOẠCH THỰC HIỆN TIỂU LUẬN, BÀI TẬP LỚN, ĐỒ ÁN/DỰ ÁN

Tên lớp: 20242IT6094003

Khoá: 18

Họ và tên sinh viên: Trịnh Gia Khánh

Tên chủ đề: Tìm hiểu các thuật toán tìm kiếm mù và ứng dụng vào bài toán tic tac toe

Tuần	Người thực hiện	Nội dung công việc	Phương pháp thực hiện
1	Trịnh Gia Khánh	Viết lời mở đầu, lời cảm ơn, trình bày về không gian trạng thái	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
2	Trịnh Gia Khánh	Trình bày về thuật toán AT	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
3	Trịnh Gia Khánh	Tham gia Mô tả không gian trạng thái của bài toán được chọn và toán tử chuyển trạng thái	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
4	Trịnh Gia Khánh	Tham gia khởi tạo các biến cần thiết và xây dựng thuật toán tìm kiếm DFS	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
5	Trịnh Gia Khánh	Đóng góp sửa đổi bổ sung Lập trình để giải quyết bài toán đề ra	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
6	Trịnh Gia Khánh	Cho ý kiến, sửa đổi để hoàn thiện báo cáo và các biểu mẫu, phiếu liên quan	Tổng hợp tất cả các nội dung báo cáo, các biểu mẫu, phiếu liên quan

Ngày tháng năm 2025

XÁC NHẬN CỦA GIẢNG VIÊN

(Ký, ghi rõ họ tên)

KẾ HOẠCH THỰC HIỆN TIỂU LUẬN, BÀI TẬP LỚN, ĐỒ ÁN/DỰ ÁN

Tên lớp: 20242IT6094003

Khoá: 18

Họ và tên sinh viên: Trương Công Minh

Tên chủ đề: Tìm hiểu các thuật toán tìm kiếm mù và ứng dụng vào bài toán tic tac toe

Tuần	Người thực hiện	Nội dung công việc	Phương pháp thực hiện
1	Trương Công Minh	Trình bày thuật toán tìm kiếm mù	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
2	Trương Công Minh	Trình bày về thuật toán AKT	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
3	Trương Công Minh	Tham gia mô tả không gian trạng thái của bài toán được chọn và chuyển đổi không gian trạng thái thành đồ thị	Nghiên cứu, tìm hiểu và tra cứu tài liệu tham khảo
4	Trương Công Minh	Tham gia vào việc khởi tạo các biến cần thiết và xây dựng thuật toán tìm kiếm DFS	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
5	Trương Công Minh	Tham gia lập trình để giải quyết bài toán đề ra	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
5	Trương Công Minh	Tham gia hoàn thiện báo cáo và các biểu mẫu, phiếu liên quan	Tổng hợp tất cả các nội dung báo cáo, các biểu mẫu, phiếu liên quan

Ngày tháng năm 2025

XÁC NHẬN CỦA GIẢNG VIÊN

(Ký, ghi rõ họ tên)

KẾ HOẠCH THỰC HIỆN TIỂU LUẬN, BÀI TẬP LỚN, ĐỒ ÁN/DỰ ÁN

Tên lớp: 20242IT6094003

Khoá: 18

Họ và tên sinh viên: Trần Thị Thảo Ngân

Tên chủ đề: Tìm hiểu các thuật toán tìm kiếm mù và ứng dụng vào bài toán tic tac toe

Tuần	Người thực hiện	Nội dung công việc	Phương pháp thực hiện
1	Trần Thị Thảo Ngân	Trình bày về không gian trạng thái	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
2	Trần Thị Thảo Ngân	Trình bày về thuật toán A*	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
3	Trần Thị Thảo Ngân	Tham gia mô tả không gian trạng thái của bài toán được chọn và chuyển đổi không gian trạng thái thành đồ thị	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
4	Trần Thị Thảo Ngân	Tham gia khởi tạo các biến cần thiết và xây dựng thuật toán tìm kiếm DFS	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
5	Trần Thị Thảo Ngân	Tham gia lập trình để giải quyết bài toán đề ra	Nghiên cứu, tìm hiểu, áp dụng những phần đã học và tra cứu tài liệu tham khảo
6	Trần Thị Thảo Ngân	Làm báo cáo và các biểu mẫu, phiếu liên quan	Tổng hợp tất cả các nội dung báo cáo, các biểu mẫu, phiếu liên quan

Ngày tháng năm 2025

XÁC NHẬN CỦA GIẢNG VIÊN

(Ký, ghi rõ họ tên)

BÁO CÁO HỌC TẬP CÁ NHÂN

Tên lớp : 2024IT6094003

Khóa : 18

Họ và tên sinh viên: Trần Đào Duy Anh

Tên chủ đề: Tìm hiểu các thuật toán tìm kiếm mù và ứng dụng vào bài toán tic tac toe

Tuần	Người thực hiện	Nội dung công việc	Kết quả đạt được	Kiến nghị với giảng viên hướng dẫn (Nêu những khó khăn, hỗ trợ từ phía giảng viên, ... nếu cần)
1	Trần Đào Duy Anh	Tham gia, đóng góp lựa chọn đề tài. Trình bày về thuật toán tìm kiếm mù	Đề tài: Tìm hiểu thuật toán tìm kiếm mù và ứng dụng vào bài toán tic tac toe. Hoàn thành nhiệm vụ.	
2	Trần Đào Duy Anh	Trình bày về tìm kiếm tối ưu	Hoàn thành nhiệm vụ	
3	Trần Đào Duy Anh	Mô tả không gian trạng thái của bài toán được chọn và toán tử chuyển trạng thái	Hoàn thành nhiệm vụ	
4	Trần Đào Duy Anh	Khởi tạo các biến cần thiết và xây dựng thuật toán tìm kiếm DFS	Hoàn thành việc khởi tạo các biến cần thiết và xây dựng thuật toán tìm kiếm DFS theo kế hoạch dự kiến.	

5	Trần Đào Duy Anh	Tham gia sửa đổi bổ sung lập trình giải quyết bài toán	Hoàn thành việc lập trình để giải quyết bài toán theo kế hoạch dự kiến.	
6	Trần Đào Duy Anh	Hoàn thiện báo cáo và các biểu mẫu, phiếu liên quan	Hoàn thiện bài báo cáo theo kế hoạch dự kiến.	

Ngày tháng năm 2025

XÁC NHẬN CỦA GIẢNG VIÊN

(Ký, ghi rõ họ tên)

BÁO CÁO HỌC TẬP CÁ NHÂN

Tên lớp : 2024IT6094003

Khóa : 18

Họ và tên sinh viên: Trịnh Gia Khánh

Tên chủ đề: Tìm hiểu các thuật toán tìm kiếm mù và ứng dụng vào bài toán tic tac toe

Tuần	Người thực hiện	Nội dung công việc	Kết quả đạt được	Kiến nghị với giảng viên hướng dẫn (Nêu những khó khăn, hỗ trợ từ phía giảng viên, ... nếu cần)
1	Trịnh Gia Khánh	Tham gia, đóng góp lựa chọn đề tài. Viết Lời mở đầu, lời cảm ơn. Chương 1 làm phần Mô tả trạng thái.	Đề tài: Tìm hiểu thuật toán tìm kiếm mù và ứng dụng vào bài toán tic tac toe. Hoàn thành nhiệm vụ chương 1 theo kế hoạch dự kiến.	
2	Trịnh Gia Khánh	Chương 1 làm phần Trình bày về thuật toán AT.	Hoàn thành nhiệm vụ chương 1 theo kế hoạch dự kiến.	
3	Trịnh Gia Khánh	Chương 2 làm phần Mô tả không gian trạng thái của bài toán được chọn và toán tử chuyển trạng thái	Hoàn thành nhiệm vụ chương 2 theo kế hoạch dự kiến.	
4	Trịnh Gia Khánh	Sử dụng ngôn ngữ Python để áp dụng vào bài toán đã đề ra theo phương pháp tìm kiếm DFS	Hoàn thành việc khởi tạo các biến cần thiết và xây dựng thuật toán tìm kiếm DFS	

			theo kế hoạch dự kiến.	
5	Trịnh Gia Khánh	Tham gia sửa đổi, bổ sung lập trình giải quyết bài toán	Hoàn thành việc lập trình để giải quyết bài toán theo kế hoạch dự kiến.	
6	Trịnh Gia Khánh	Tham gia sửa đổi, bổ sung, hoàn thiện bài tập lớn của nhóm.	Hoàn thiện bài báo cáo theo kế hoạch dự kiến.	

Ngày tháng năm 2025

XÁC NHẬN CỦA GIẢNG VIÊN

(Ký, ghi rõ họ tên)

BÁO CÁO HỌC TẬP CÁ NHÂN

Tên lớp : 2024IT6094003

Khóa : 18

Họ và tên sinh viên: Trương Công Minh

Tên chủ đề: Tìm hiểu các thuật toán tìm kiếm mù và ứng dụng vào bài toán tic tac toe

Tuần	Người thực hiện	Nội dung công việc	Kết quả đạt được	Kiến nghị với giảng viên hướng dẫn (Nêu những khó khăn, hỗ trợ từ phía giảng viên, ... nếu cần)
1	Trương Công Minh	Trình bày thuật toán tìm kiếm mù	Hoàn thành đúng thời hạn và đầy đủ nội dung	
2	Trương Công Minh	Trình bày về thuật toán AKT	Hoàn thành đúng thời hạn và đầy đủ nội dung	
3	Trương Công Minh	Tham gia mô tả không gian trạng thái của bài toán được chọn và toán tử chuyển trạng thái	Hoàn thành đúng thời hạn và đầy đủ nội dung	
4	Trương Công Minh	Tham gia vào việc khởi tạo các biến cần thiết và xây dựng thuật toán tìm kiếm DFS	Hoàn thành đúng thời hạn và đầy đủ nội dung	
5	Trương Công Minh	Tham gia lập trình để giải quyết bài toán đề ra	Hoàn thành đúng thời hạn và đầy đủ nội dung	
6	Trương Công Minh	Tham gia hoàn thiện báo cáo và	Hoàn thành đúng thời hạn và đầy đủ nội dung	

		các biểu mẫu, phiếu liên quan.		
--	--	-----------------------------------	--	--

Ngày tháng năm 2025

XÁC NHẬN CỦA GIẢNG VIÊN

(Ký, ghi rõ họ tên)

BÁO CÁO HỌC TẬP CÁ NHÂN

Tên lớp : 2024IT6094003

Khóa : 18

Họ và tên sinh viên: Trần Thị Thảo Ngân

Tên chủ đề: Tìm hiểu các thuật toán tìm kiếm mù và ứng dụng vào bài toán tic tac toe

Tuần	Người thực hiện	Nội dung công việc	Kết quả đạt được	Kiến nghị với giảng viên hướng dẫn (Nêu những khó khăn, hỗ trợ từ phía giảng viên, ... nếu cần)
1	Trần Thị Thảo Ngân	Tham gia, đóng góp lựa chọn đề tài. Trình bày về không gian trạng thái	Đề tài: Tìm hiểu thuật toán tìm kiếm mù và ứng dụng vào bài toán tic tac toe. Hoàn thành đúng thời hạn và đầy đủ nội dung	
2	Trần Thị Thảo Ngân	Trình bày về thuật toán A*	Hoàn thành đúng thời hạn và đầy đủ nội dung	
3	Trần Thị Thảo Ngân	Tham gia mô tả không gian trạng thái của bài toán được chọn và toán tử chuyển trạng thái	Hoàn thành đầy đủ nội dung	
4	Trần Thị Thảo Ngân	Tham gia khởi tạo các biến cần thiết và xây dựng thuật toán tìm kiếm DFS	Thống nhất về các biến và xây dựng được thuật toán	

5	Trần Thị Thảo Ngân	Tham gia lập trình để giải quyết bài toán đề ra	Lựa chọn ngôn ngữ lập trình Python, sử dụng các biến và thuật toán đã xây dựng để hoàn thành sản phẩm. Kết quả chạy thử ra đúng đáp án	
6	Trần Thị Thảo Ngân	Tham gia sửa đổi, làm báo cáo và các biểu mẫu, phiếu liên quan	Hoàn thành đầy đủ và đúng thời hạn	

Ngày tháng năm 2025

XÁC NHẬN CỦA GIẢNG VIÊN

(Ký, ghi rõ họ tên)

PHIẾU HỌC TẬP CÁ NHÂN

I. Thông tin chung

1. Tên lớp: 2024IT6094003 Khóa: 18
2. Họ và tên sinh viên: Trần Đào Duy Anh

II. Nội dung học tập

1. Tên chủ đề: Tìm hiểu các thuật toán tìm kiếm mù và ứng dụng vào bài toán tic tac toe
2. Hoạt động của sinh viên
 - *Hoạt động 1: Đề xuất chủ đề nghiên cứu*
 - + Nội dung:
 - Tham gia viết đề xuất lựa chọn chủ đề nghiên cứu
 - Tham gia lập biên bản họp và làm việc nhóm (BM03)
 - Tham gia, đóng góp đặt ra các quy tắc làm việc nhóm:
 - Các thành viên cần tham gia đầy đủ và đúng giờ các cuộc họp
 - Vắng mặt phải báo trước với nhóm trưởng trước buổi họp ít nhất 12 tiếng và có lý do chính đáng
 - Mỗi thành viên có trách nhiệm hoàn thành công việc đã được giao nghiêm túc và đúng hạn
 - Có vướng mắc thì trao đổi với nhóm trưởng và các thành viên trong nhóm qua nhóm Zalo. Nếu không tìm được hướng giải quyết thì hỏi giảng viên hướng dẫn – Cô Mai Thanh Hồng
 - + Mục tiêu/chuẩn đầu ra: Lập biên bản họp và làm việc nhóm (BM03)
 - *Hoạt động 2: Báo cáo tiến độ lần 1*
 - + Nội dung:
 - Trình bày tổng quan về các thuật toán tìm kiếm mù
 - + Mục tiêu/chuẩn đầu ra: Biết được thuật toán tìm kiếm mù gồm những thuật toán nào, phục vụ làm nền tảng cho các bước nghiên cứu sau.
 - *Hoạt động 3: Báo cáo tiến độ lần 2*
 - + Nội dung:
 - Giới thiệu tổng quan tìm kiếm tối ưu
 - Trình bày các ứng dụng điển hình của tìm kiếm mù
 - + Mục tiêu/chuẩn đầu ra: Hiểu được cách hoạt động, vai trò và ứng dụng của tìm kiếm mù, giải thích nguyên lý.

- *Hoạt động 4: Mô tả không gian trạng thái của bài toán được chọn và toán tử chuyển trạng thái*

+ Nội dung:

- Xác định và mô tả các thành phần cơ bản của không gian trạng thái: trạng thái ban đầu, trạng thái đích, tập hợp các hành động, và các ràng buộc nếu có.
- Minh họa không gian trạng thái bằng sơ đồ, bảng hoặc mô hình trực quan.
- Phân tích mối quan hệ giữa các trạng thái và cách chuyển đổi từ trạng thái này sang trạng thái khác.

+ Mục tiêu: Trình bày được đầy đủ, rõ ràng không gian trạng thái của bài toán được chọn.

- *Hoạt động 5: Khởi tạo các biến cần thiết và xây dựng thuật toán tìm kiếm DFS*

+ Nội dung:

- Giới thiệu tổng quan về thuật toán tìm kiếm DFS
- Trình bày các ứng dụng điển hình của thuật toán

+ Mục tiêu:

- *Hoạt động 6: Hoàn thiện báo cáo và các biểu mẫu, phiếu liên quan*

+ Nội dung:

- Xem những lỗi sai và thiếu sót để sửa
- Đóng góp ý kiến bổ sung

+ Mục tiêu/chuẩn đầu ra: Hoàn thiện và nộp sản phẩm nghiên cứu

- 3. Sản phẩm nghiên cứu: Quyền báo cáo thí nghiệm/thực nghiệm + Chương trình code

III. Nhiệm vụ học tập

1. Hoàn thành báo cáo thí nghiệm/thực nghiệm theo đúng thời gian quy định (từ ngày 21/04/2025 đến ngày 25/05/2025)
2. Báo cáo sản phẩm nghiên cứu theo chủ đề được giao khi kết thúc thời gian thực hiện.

IV. Học liệu thực hiện Bài tập lớn

1. Tài liệu học tập

[1] Nguyễn Phương Nga, Trần Hùng Cường, *Giáo trình Trí tuệ nhân tạo*, NXB Thống kê, 2021.

[2] Nguyễn Thanh Thủy, *Trí tuệ nhân tạo*, NXB Thống kê, 1999.

[3] Nguyễn Đình Thức, *Giáo trình Trí tuệ nhân tạo: Mạng neuron phương pháp và ứng dụng*, NXB Giáo dục, 2000.

Phương tiện, nguyên liệu thực hiện báo cáo: Máy tính cá nhân, mạng internet.

PHIẾU HỌC TẬP CÁ NHÂN

I. Thông tin chung

1. Tên lớp: 2024IT6094003 Khóa: 18
2. Họ và tên sinh viên: Trịnh Gia Khánh

II. Nội dung học tập

1. Tên chủ đề: Tìm hiểu các thuật toán tìm kiếm mù và ứng dụng vào bài toán tic tac toe

2. Hoạt động của sinh viên

- *Hoạt động 1: Đề xuất chủ đề nghiên cứu*

+ Nội dung:

- Tham gia viết đề xuất lựa chọn chủ đề nghiên cứu
- Tham gia lập biên bản họp và làm việc nhóm (BM03)
- Tham gia, đóng góp đặt ra các quy tắc làm việc nhóm:
 - Các thành viên cần tham gia đầy đủ và đúng giờ các cuộc họp
 - Vắng mặt phải báo trước với nhóm trưởng trước buổi họp ít nhất 12 tiếng và có lý do chính đáng
 - Mỗi thành viên có trách nhiệm hoàn thành công việc đã được giao nghiêm túc và đúng hạn
 - Có vướng mắc thì trao đổi với nhóm trưởng và các thành viên trong nhóm qua nhóm Zalo. Nếu không tìm được hướng giải quyết thì hỏi giảng viên hướng dẫn – Cô Mai Thanh Hồng

+ Mục tiêu/chuẩn đầu ra: Lập biên bản họp và làm việc nhóm (BM03)

- *Hoạt động 2: Báo cáo tiến độ lần 1*

+ Nội dung:

- Viết lời mở đầu, lời cảm ơn
- Trình bày về không gian trạng thái
- Trình bày về thuật toán AT

+ Mục tiêu/chuẩn đầu ra: Viết được nội dung Chương 1: Không gian trạng thái và các thuật toán tìm kiếm: Giới thiệu tổng quan về không gian trạng thái, toán tử chuyển trạng thái, thuật toán Heuristic.

- *Hoạt động 3: Báo cáo tiến độ lần 2*

+ Nội dung:

- Mô tả không gian trạng thái của bài toán được chọn và toán tử chuyển trạng thái
- Khởi tạo các biến cần thiết và xây dựng thuật toán tìm kiếm DFS
- Lập trình để giải quyết bài toán đề ra

+ Mục tiêu/chuẩn đầu ra: Viết được nội dung Chương 2: Xây dựng chương trình: Tiến hành mô tả bài toán đã được đưa ra và sử dụng ngôn ngữ lập trình Python để ứng dụng thuật toán tìm kiếm mù giải bài toán tic tac toe.

- *Hoạt động 4: Nộp cuốn Báo cáo thí nghiệm/thực nghiệm và chương trình code*

+ Nội dung:

- Nêu ý kiến góp ý để hoàn thiện lập trình bài toán ứng dụng
- Nêu ý kiến góp ý để hoàn thiện bài báo cáo của nhóm

+ Mục tiêu/chuẩn đầu ra: Hoàn thiện và nộp sản phẩm nghiên cứu

3. Sản phẩm nghiên cứu: Quyển báo cáo thí nghiệm/thực nghiệm + Chương trình code

III. Nhiệm vụ học tập

1. Hoàn thành báo cáo thí nghiệm/thực nghiệm theo đúng thời gian quy định (từ ngày 21/04/2025 đến ngày 25/05/2025)
2. Báo cáo sản phẩm nghiên cứu theo chủ đề được giao khi kết thúc thời gian thực hiện.

IV. Học liệu thực hiện Bài tập lớn

1. Tài liệu học tập

[1] Nguyễn Phương Nga, Trần Hùng Cường, *Giáo trình Trí tuệ nhân tạo*, NXB Thống kê, 2021.

[2] Nguyễn Thanh Thủy, *Trí tuệ nhân tạo*, NXB Thống kê, 1999.

[3] Nguyễn Đình Thức, *Giáo trình Trí tuệ nhân tạo: Mạng neuron phương pháp và ứng dụng*, NXB Giáo dục, 2000.

2. Phương tiện, nguyên liệu thực hiện báo cáo: Máy tính cá nhân, mạng internet.

PHIẾU HỌC TẬP CÁ NHÂN

I. Thông tin chung

1. Tên lớp: 2024IT6094003 Khóa: 18
2. Họ và tên sinh viên: Trương Công Minh

II. Nội dung học tập

1. Tên chủ đề: Tìm hiểu các thuật toán tìm kiếm mù và ứng dụng vào bài toán tic tac toe
2. Hoạt động của sinh viên
 - *Hoạt động 1: Đề xuất chủ đề nghiên cứu*
 - + Nội dung 1:
 - Đề xuất lựa chọn chủ đề nghiên cứu và xin ý kiến của người hướng dẫn về chủ đề nghiên cứu
 - Lập biên bản họp và làm việc nhóm tuần 1 và tuần 5 (theo BM03)
 - Tham gia, đóng góp đặt ra các quy tắc làm việc nhóm:
 - Các thành viên cần tham gia đầy đủ và đúng giờ các cuộc họp
 - Vắng mặt phải báo trước với nhóm trưởng trước buổi họp ít nhất 12 tiếng và có lý do chính đáng
 - Mỗi thành viên có trách nhiệm hoàn thành công việc đã được giao nghiêm túc và đúng hạn
 - Có vướng mắc thì trao đổi với nhóm trưởng và các thành viên trong nhóm qua nhóm Zalo. Nếu không tìm được hướng giải quyết thì hỏi giảng viên hướng dẫn – Cô Mai Thanh Hồng
 - + Mục tiêu/chuẩn đầu ra: Biết cách lập kế hoạch thực hiện bài tập lớn (theo BM02) và lập biên bản họp và làm việc nhóm (theo BM03)
 - *Hoạt động 2: Viết đề cương nghiên cứu*
 - + Nội dung 2:
 - Tham gia viết đề cương nghiên cứu
 - Bảo vệ đề cương nghiên cứu
 - + Mục tiêu/chuẩn đầu ra: Viết được đề cương nghiên cứu
 - *Hoạt động 3: Báo cáo tiến độ lần 1*
 - + Mục tiêu/chuẩn đầu ra: Trình bày tổng quan về các thuật toán tìm kiếm mù trong trí tuệ nhân tạo và thuật toán AKT
 - *Hoạt động 4: Báo cáo tiến độ lần 2*

+ Mục tiêu/chuẩn đầu ra: Mô tả không gian bài toán và toán tử chuyển trạng thái. Tham gia vào việc khởi tạo các biến cần thiết và xây dựng thuật toán tìm kiếm DFS. Tham gia lập trình để giải quyết bài toán tic tac toe.

3. Sản phẩm nghiên cứu: Quyền báo cáo thực nghiệm/thí nghiệm và chương trình code giải quyết bài toán tic tac toe.

III. Nhiệm vụ học tập

1. Hoàn thành báo cáo thí nghiệm/thực nghiệm theo đúng thời gian quy định (từ ngày 21/04/2025 đến ngày 25/05/2025)
2. Báo cáo sản phẩm nghiên cứu theo chủ đề được giao khi kết thúc thời gian thực hiện.

IV. Học liệu thực hiện Bài tập lớn

1. Tài liệu học tập

[1] Nguyễn Phương Nga, Trần Hùng Cường, *Giáo trình Trí tuệ nhân tạo*, NXB Thống kê, 2021.

[2] Nguyễn Thanh Thủy, *Trí tuệ nhân tạo*, NXB Thống kê, 1999.

[3] Nguyễn Đình Thức, *Giáo trình Trí tuệ nhân tạo: Mạng neuron phương pháp và ứng dụng*, NXB Giáo dục, 2000.

2. Phương tiện, nguyên liệu thực hiện báo cáo: Máy tính cá nhân, mạng internet.

PHIẾU HỌC TẬP CÁ NHÂN

I. Thông tin chung

1. Tên lớp: 2024IT6094003 Khóa: 18
2. Họ và tên sinh viên: Trần Thị Thảo Ngân

II. Nội dung học tập

1. Tên chủ đề: Tìm hiểu các thuật toán tìm kiếm mù và ứng dụng vào bài toán tic tac toe
2. Hoạt động của sinh viên
 - *Hoạt động 1: Đề xuất chủ đề nghiên cứu*
 - + Nội dung:
 - Tham gia viết đề xuất lựa chọn chủ đề nghiên cứu
 - Tham gia lập biên bản họp và làm việc nhóm (BM03)
 - Tham gia đặt ra các quy tắc làm việc nhóm:
 - Các thành viên cần tham gia đầy đủ và đúng giờ các cuộc họp
 - Vắng mặt phải báo trước với nhóm trưởng trước buổi họp ít nhất 12 tiếng và có lý do chính đáng
 - Mỗi thành viên có trách nhiệm hoàn thành công việc đã được giao nghiêm túc và đúng hạn
 - Có vướng mắc thì trao đổi với nhóm trưởng và các thành viên trong nhóm qua nhóm Zalo. Nếu không tìm được hướng giải quyết thì hỏi giảng viên hướng dẫn – Cô Mai Thanh Hồng
 - + Mục tiêu/chuẩn đầu ra: Lập biên bản họp và làm việc nhóm (BM03)
 - *Hoạt động 2: Báo cáo tiến độ lần 1*
 - + Nội dung:
 - Trình bày về không gian trạng thái
 - Trình bày về thuật toán A*
 - + Mục tiêu/chuẩn đầu ra: Viết được nội dung Chương 1: Không gian trạng thái và các thuật toán tìm kiếm: Giới thiệu tổng quan về không gian trạng thái, toán tử chuyển trạng thái, Thuật toán Heuristic.
 - *Hoạt động 3: Báo cáo tiến độ lần 2*
 - + Nội dung:
 - Tham gia mô tả không gian trạng thái của bài toán được chọn và toán tử chuyển trạng thái

- Khởi tạo các biến cần thiết và xây dựng thuật toán tìm kiếm DFS
- Tham gia lập trình để giải quyết bài toán đề ra

+ Mục tiêu/chuẩn đầu ra: Viết được nội dung Chương 2: Xây dựng chương trình: Tiến hành mô tả bài toán đã được đưa ra và sử dụng ngôn ngữ lập trình Python để ứng dụng thuật toán tìm kiếm mù giải bài toán tic tac toe.

- *Hoạt động 4: Nộp cuốn Báo cáo thí nghiệm/thực nghiệm và chương trình code*

+ Nội dung:

- Nêu ý kiến góp ý để hoàn thiện lập trình bài toán ứng dụng
- Nêu ý kiến góp ý để hoàn thiện bài báo cáo của nhóm

+ Mục tiêu/chuẩn đầu ra: Hoàn thiện và nộp sản phẩm nghiên cứu

3. Sản phẩm nghiên cứu: Quyển báo cáo thí nghiệm/thực nghiệm + Chương trình code

III. Nhiệm vụ học tập

1. Hoàn thành báo cáo thí nghiệm/thực nghiệm theo đúng thời gian quy định (từ ngày 21/04/2025 đến ngày 25/05/2025)
2. Báo cáo sản phẩm nghiên cứu theo chủ đề được giao khi kết thúc thời gian thực hiện.

IV. Học liệu thực hiện Bài tập lớn

1. Tài liệu học tập

[1] Nguyễn Phương Nga, Trần Hùng Cường, *Giáo trình Trí tuệ nhân tạo*, NXB Thống kê, 2021.

[2] Nguyễn Thanh Thủy, *Trí tuệ nhân tạo*, NXB Thống kê, 1999.

[3] Nguyễn Đình Thức, *Giáo trình Trí tuệ nhân tạo: Mạng neuron phương pháp và ứng dụng*, NXB Giáo dục, 2000.

2. Phương tiện, nguyên liệu thực hiện báo cáo: Máy tính cá nhân, mạng internet.