TRƯỜNG ĐẠI HỌC KINH DOANH VÀ CÔNG NGHỆ HÀ NỘI

KHOA CÔNG NGHỆ THÔNG TIN



KẾ HOẠCH VÀ BÁO CÁO TIẾN ĐỘ

BÀI TẬP KHỞI NGHIỆP MÔN TRÍ TUỆ NHÂN TẠO

**ĐỀ TÀI: THUẬT TOÁN AKT MINH HOẠ TRONG BÀI TOÁN TACI**

**XÂY DỰNG GAME GHÉP TRANH 8 Ô SỐ**

**Nhóm: 5 - Lớp: TH24.25**

Nguyễn Anh Quốc

Nguyễn Văn Quỳnh

Hoàng Mạnh Sỹ

Đặng Xuân Thắng

Nguyễn Thị Thùy

**Hà Nội – 12/2021**

**KẾ HOẠCH THỰC HIỆN**

**BẢNG PHÂN CÔNG CÔNG VIỆC CHO THÀNH VIÊN**

|  |  |  |
| --- | --- | --- |
| **TT** | **Họ tên** | **Nội dung công việc được phân công** |
| 1 | Nguyễn Anh Quốc | Phần kết luận |
| 2 | Nguyễn Văn Quỳnh | Tổng quan và mô tả bài toán thực tế |
| 3 | Hoàng Mạnh Sỹ | Game ghép tranh 8 số, hoàn thành báo cáo |
| 4 | Đặng Xuân Thắng | Phần mở đầu |
| 5 | Nguyễn Thị Thùy | Tổng quan và mô tả bài toán thực tế |

# NỘI DUNG CÔNG VIỆC ĐÃ THỰC HIỆN ĐƯỢC (BUỔI 1)

|  |  |  |
| --- | --- | --- |
| **TT** | **Nội dung công việc đã thực hiện được** | **Ghi chú** |
| 1 | Phần mở đầu | Hoàn thành |
| 2 | Tổng quan và mô tả bài toán thực tế | Hoàn thành |
| 3 | Game ghép tranh 8 số | Hoàn thành |
| 4 | Phần kết luận | Hoàn thành |

# ĐÁNH GIÁ THÀNH VIÊN TRONG NHÓM ĐÃ THỰC HIỆN

# (Trưởng nhóm đánh giá trong buổi 1)

|  |  |  |
| --- | --- | --- |
| **TT** | **Họ và tên** | **Hệ số** |
| 1 | Nguyễn Anh Quốc | 0.5 |
| 2 | Nguyễn Văn Quỳnh | 0.5 |
| 3 | Hoàng Mạnh Sỹ | 1 |
| 4 | Đặng Xuân Thắng | 0.7 |
| 5 | Nguyễn Thị Thùy | 1 |

## Chương 1 : Giới thiệu và mô tả bài toán thực tế

## Lý do chọn đề tài

Trong thời đại công nghệ 4.0 đang ngày càng phát triển, ứng dụng trí tuệ nhân tạo vào đời sống được xem là một trong những xu thế phát triển mạnh mẽ hiện nay. Việc tự động hóa các hành vi thông minh đang trở thành một cuộc cách mạng trong ngành CNTT.

Bài toán ghép , hay với các tên gọi khác như "Gem puzzle", "Mystic Square" nằm trong nhóm các bài toán puzzle (xếp hình, đẩy hộp đại loại). Trò chơi này gồm một bảng có n - 1 ô vuông (phổ biến n = 8, 15, 24, ...) tức là sẽ có một ô trống trong bảng. Các ô liền kề có để di chuyển đến ô trống này.  Từ một trạng thái bất kì, người chơi cần di chuyển các ô của bảng về một trạng thái đích nào đó với số lần di chuyển càng ít càng tốt. Về thuật toán giản bài toán này, có khá là nhiều cách:

* Breadth-first Search (BFS)
* Depth-first Search (DFS)
* Iterative Deepening DFS (ID-DFS)
* Greedy Search
* AKT Search, A\* Search

Chính vì vậy thông qua việc học môn trí tuệ nhân tạo (AI) nhóm em đã nghĩ ra một ý tưởng nhỏ đó là áp dụng thuật toán mình đã học để làm ra một Game xếp hình 8 ô số. Phục vụ cho việc chứng minh áp dụng trí tuệ nhân tạo mang lại lợi ích tối ưu về không gian và thời gian cho con người.

## Mô tả bài toán thực tế được giải quyết

* Vị trí của các hình trong trò chơi sẽ nằm ngẫu nhiên trộn lẫn trong 9 ô, trong đó có 1 ô đen để người dùng dịch chuyển đi từng bước. Mỗi lần di chuyển người dùng chỉ có thể đi 1 bước theo chiều qua trái, qua phải, đi lên hoặc đi xuống để ghép thành 1 hình hoàn chỉnh theo hình mẫu đã cho theo đó. Người dùng **không** được đi chéo.
* Trong quá trình chơi có thể có trường hợp người dùng không thể đi đến trạng thái hoàn chỉnh của hình. Vì vậy chúng ta áp dụng trí tuệ nhân tạo vào trong trò chơi này, và cụ thể là chúng em đã áp dụng 2 thuật toán AKT trong game này để đưa người chơi đi đến trạng thái hoàn chỉnh (giải ra được đường đi đến trạng thái hoàn chỉnh).
* Vậy bài toán thực tế ở đây là tìm ra đường đi đến trạng thái hoàn thành của game ghép tranh 8 ô số, cũng như đưa một trạng thái ngẫu nhiên chưa hoàn chỉnh của bức tranh về một bức tranh hoàn chỉnh áp dụng 2 thuật toán trong trí tuệ nhân tạo.

# Chương 2 : Tổng quan về Thuật giải AKT , Áp dụng bài toán TACI

## 2.1 Tổng quan về thuật giải AKT` (Algorithm for Knowlegeable Tree Search) Thuật giải AKT mở rộng AT bằng cách sử dụng thêm thông tin ước lượng h’. Độ tốt của một trạng thái f là tổng của hai hàm g và h’.

Begin   
 open:={s};   
While(open<>0)do   
 begin   
 n:= Retrieve(Open) //Chọn n sao cho f(n) →nhỏ nhất từ Open.   
 if(n=g) then return True   
 else begin  
 Tạo F(n)   
 for mỗi nút con m của F(n) do   
 Begin  
 g(m):=g(n)+Cost(n,m)  
 f(m):=g(m)+h’(m);   
 Open:=Open{m};   
 end;   
 end;   
 Return False;   
End;

## 2.2 Trình bày thuật giải Bước 1:

* Mọi đỉnh n và hàm g,h, f ẩn.
* Mở đỉnh đầu tiên S0 . Gán g(S0) =0
* Sử dụng tri thức bổ sung ước tính h(S0)
* Tính f(S0) = g(S0) + h(S0)

Bước 2:

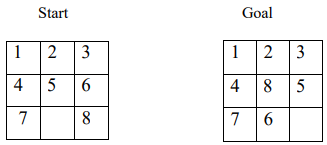
* Chọn một đỉnh mở với hàm f là min và gọi là đỉnh N.
* Nếu N là đích => dừng. Đường đi từ đỉnh ban đầu đến đỉnh N là ngắn nhất và bằng g(N).
* Nếu không tồn tại N thì cây biểu diễn vấn đề không có đường đi tới mục tiêu => dừng (bài toán không lời giải)
* Nếu tồn tại nhiều hơn một đỉnh N có cùng hàm fmin thì phải kiểm tra xem trong số đó có đỉnh nào là đích hay không?
  + Nếu có => Dừng
  + Nếu không=>chọn ngẫu nhiên một trong các đỉnh đó và gọi đó là đỉnh N

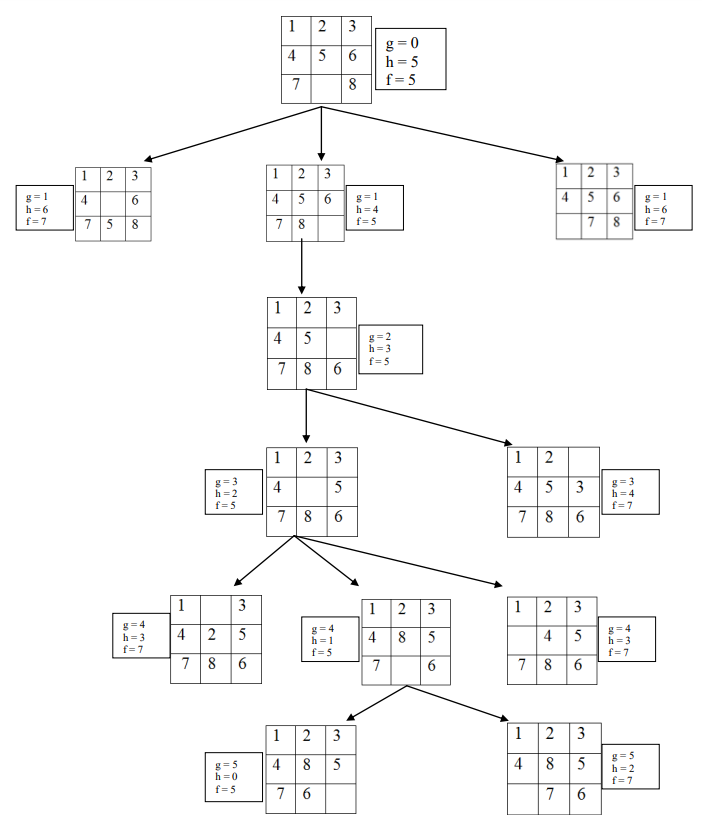
Bước 3:

* Đóng đỉnh N và mở mọi đỉnh sau N.
* Với mọi đỉnh S sau N, tính g(s)=g(N) + g(N ->S)
* Dùng tri thức bổ sung để ước tính hàm h(S)
* Tính f(S) = g(S) + h(S)

Bước 4: Quay lại bước 2**2.3 Áp dụng bài toán TACI**

Bài toán TACI: có n2-1 số mang các giá trị từ 1 tới n2-1 được sắp xếp vào một lưới các ô vuông kính thưc. Mỗi số đó được gọi là một quân cơ và lưới ô đó được gọi là bàn cờ, có một vị trí bàn cờ đó bỏ trống. Mỗi lần di chuyển quân người chơi được phép di chuyển một quân ở vị trí ô tiếp giáp cạnh với ô trống vào ô trống.  
 Yêu cầu : từ trạng thái ban đầu (a), thực hiện các nước đi hợp lệ để thu được trang thái kết thúc (b)



Ta có:

Vì h=0 => đích => dừng

Kết luận: đường đi ngắn nhất g=5

## 2.3. Bài toán ghép tranh

Game ghép tranh(N-Puzzle) là một trò chơi khá hay và trí tuệ, nó được biết đến với nhiều phiên bản và tên gọi khác nhau như: 8-puzzle, 15-puzzle, Gem puzzle, Boss puzzle. Bài toán N-puzzle là vấn đề cổ điển cho mô hình thuật toán liên quan đến trí tuệ nhân tạo. Bài toán đặt ra là phải tìm đường đi từ trạng thái hiện tại tới trạng thái đích. Và cho tới nay vẫn chưa có thuật toán tối ưu để giải bài toán này.   
Khi chơi người dùng có thể sử dụng chuột/bàn phím chơi với các kích thước khác nhau và với hình ảnh khác nhau hoặc có thể sử dụng chức năng tìm lời giải nhờ thuật toán AKT.

 Yêu cầu xây dựng bảng ô vuông n hàng, n cột. Bảng gồm 1 ô trống và n2-1 ô chứa các số trong phạm vi [1, n2-1]. Xuất phát từ một cách xếp bất kì, di chuyển ô trống lên trên, xuống dưới, sang phải, sang trái để đưa các ô về trạng thái đích. Sử dụng chuột hay phím chức năng để di chuyển ô trống. Chương trình có chức năng tự động chơi ở bất kì trạng thái nào đó. Mỗi trạng thái của bảng số là một hoán vị của n2 phần tử. Ở đây ta có thể mở rộng bằng việc thêm hình ảnh vào game hoặc gắn số vào hình ảnh để gợi ý cho người chơi. Ở trạng thái ban đầu, các ô được sắp xếp ngẫu nhiên, và nhiệm vụ của người chơi là tìm được cách đưa chúng về trạng thái đích(ô đầu trống, các ô khác theo thứ tự tăng dần từ trái qua phải, từ trên xuống dưới). Để đơn giản trong cách tiếp cận bài toán, ta giả định chỉ các ô trống di chuyển trong bảng là di chuyển đến các vị trí khác nhau. Như vậy tại một trạng thái bất kì có tối đa 4 cách di chuyển đến trạng thái khác(trái, phải, lên, xuống).



 Bắt đầu Kết thúc

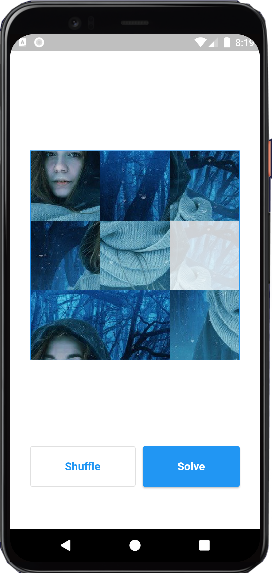
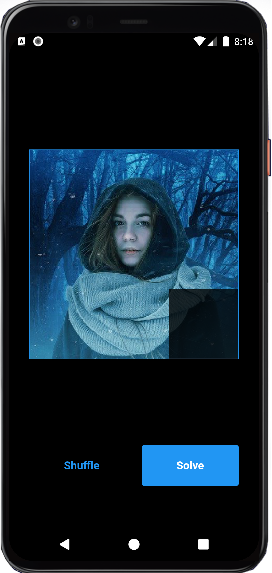
Bước di chuyển của ô trống:



Dưới Trên Phải Trái

**Chương 3 : Các chức năng chính của Game**

## 3.1 Tổng quan về chức năng, giao diện



Giao diện sáng Giao diện tối

**3.2 Các chức năng chính trong game**

3.2.1 Chức năng Shuffle

Khi nhấn nút Shuffle, thứ tự tranh sẽ được thay đổi tự động và ngẫu nhiên

3.2.2 Chức năng Solve

Khi nhấn nút Solve, ứng dựng sẽ chạy thuật toán, giải quyết bài toán và đưa thứ tự tranh về đúng vị trí của nó

**3.3 Cách chơi**

Ban đầu người chơi có thể nhấn nút Shuffle để ứng dụng tự động xáo trộn tranh theo thứ tự ngẫu nhiên. Hoặc cũng có thể tự di chuyển tranh theo ý của mình, bằng cách nhấn vào những bức tranh ở cạnh ô trống, chúng sẽ đổi chỗ cho ô trống

Sau khi đã xáo trộn tranh, người dung có thể tự giải trò chơi, hoặc có thể bấm nút Solve để ứng dụng khởi chạy thuật toán, tự động đưa tranh về đúng vị trí ban đầu

# **Chương 4: Kết Luận**

# **4**.1 Kết luận

## Thuật toán AKT là giải thuật tìm kiếm đường đi tốt nhất đối với cây chỉ có thông tin về đỉnh, cung và giá trị của cung.

* Thuật toán AKT được ứng dụng vào nhiều bài toán, ví dụ như bài toán TACI, bài toán tháp Hà Nội,….
* Áp dụng thuật toán AKT vào nhưunxg công việc cần có độ chính xác cao và cần có thời gian tối ưu nhất.

## 4.2 Kết quả đạt được

* Tìm hiểu được phần cơ bản của thuật toán A**KT**
* Xây dựng thành công game ghép tranh , áp dụng các giải tuật phù hợp
* Hiểu được cấu trúc, cách thức hoạt động, cách thức phát triển áp dụng các thuật toán vào xây dựng game
* Nâng cao kiến thức lập trình và tiếng anh chuyên ngành thông qua những tài liệu về trí tuệ nhân tạo bằng tiếng anh

**4.3 Hạn chế định hướng phát triển**

* Hạn chế:
  + Do kiến thức của về Trí tuệ nhân tạo còn hạn hẹp nên chưa tìm hiểu kĩ, chuyên và nâng cao về thuật toán AKT
  + Ứng dụng trò chơi giải tranh 8 số còn đơn giản, không có nhiều tính năng
* Định hướng phát triển:
  + Tìm hiểu sâu về Trí tuệ nhân tạo, bổ sung them kiến thức về thuật toán AKT cũng như tìm tòi, học hỏi them nhiều thuật toán khác.
  + Phát triển them các tính năng mới cho game giải tranh 8 số như tính thời gian, tính số bước đi