C:\Users\tdqua_000\Dropbox\SS-Slides\DeCuong-CDIO\Template CDIO v4.2\Templates\Hinh anh\LogoTruong.png

**BÁO CÁO PROJECT 1**

**1612272- TRẦN NHẬT HUY**

**1612282-TRẦN ĐÌNH KHẢI**

**NHẬP MÔN CƠ SỞ TRÍ TUỆ NHÂN TẠO**

**Contents**

[**1. THÔNG TIN NHÓM VÀ BÀI TOÁN 2**](#_Toc526850849)

[**1.1. Thông tin nhóm 2**](#_Toc526850850)

[**1.2. Tóm tắt bài toán: Tìm đường đi: 2**](#_Toc526850851)

[**2. PHÂN CÔNG CÔNG VIỆC 3**](#_Toc526850852)

[**3. SƠ ĐỒ BIỂU DIỄN HỆ THỐNG 4**](#_Toc526850853)

[**4. MÔ TẢ CẤU TRÚC DỮ LIỆU 4**](#_Toc526850854)

[**5. MÔ TẢ THUẬT TOÁN 4**](#_Toc526850855)

[**6. CÁC BỘ TEST 4**](#_Toc526850856)

[**7. MỨC ĐỘ HOÀN THÀNH VÀ ĐÁNH GIÁ 5**](#_Toc526850857)

# THÔNG TIN NHÓM VÀ BÀI TOÁN

|  |  |  |  |
| --- | --- | --- | --- |
| **MSSV** | **HỌ VÀ TÊN** | **EMAIL** | **SỐ ĐIỆN THOẠI** |
| 1612272 | TRẦN NHẬT HUY | [nhathuy13598@gmail.com](mailto:nhathuy13598@gmail.com) | 0354 878 677 |
| 1612282 | TRẦN ĐÌNH KHẢI | trandinhkhai10.1@gmail.com | 01679969653 |

## Thông tin nhóm

## Tóm tắt bài toán: Tìm đường đi:

Không gian di chuyển được biểu diễn thành ma trận vuông NxN, mỗi ô có một giá trị 0 là ô trống, hoặc 1 là chướng ngại vật

**O**

**y**

**x**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |

Minh hoạ với ma trận 5x5, các ô đen là chướng ngại vật

Các vị trí được xác định bởi (x,y) theo hệ toạ độ ở hình trên bắt đầu từ (0,0). Cho điểm đầu với điểm đích. Sử dụng thuật toán tìm kiếm A\* với h(n) là khoảng cách Euclidean tìm đường đi ngắn nhất <né chướng ngại vật>. Thứ tự mở của ô đang xét:

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 8 |  | 4 |
| 7 | 6 | 5 |

# PHÂN CÔNG CÔNG VIỆC

|  |  |  |
| --- | --- | --- |
| **Công việc** | **Người thực hiện** | **Mục tiêu/ sản phẩm** |
| Viết báo cáo | Khải | File báo cáo |
| Vẽ sơ đồ hệ thống | Huy | Ảnh sơ đồ |
| Vẽ cấu trúc dữ liệu | Huy | Ảnh UML |
| Viết mô tả thuật toán | Khải | Văn bản trong phần báo cáo |
| Xây dựng các bộ test | Huy | Các file input và output tương ứng |
| Code thuật toán | Khải, Huy | Source code |

# SƠ ĐỒ BIỂU DIỄN HỆ THỐNG

A(start:Node,goal:Node,matrix:list)

subNode(self:Node,data:list)

heuristic(self:Node,g:Node)

|  |  |
| --- | --- |
| Tên hàm | Công dụng |
| heuristic(self,g) | Tính khoảng cách Euclid từ 1 đỉnh đến đỉnh goal |
| subNode(self,data) | Mở rộng các node kề cận với node self. Node này chưa được đi qua và không là chướng ngại vật trong mảng các node data |
| A(start:Node,goal:Node,matrix:list) | Thuật toán A\* tìm đường đi từ start đến goal với mảng chứa các đỉnh matrix |

# MÔ TẢ CẤU TRÚC DỮ LIỆU

|  |  |
| --- | --- |
|  | Class Node |
| x | Hoành độ một node |
| y | Tung độ một node |
| g | Chi phí = heuristic + chi phí đi |
| stt | Số thứ tự mở |
| parent | Node cha |

**struct Node{**

**x,**

**y,**

**heuristic,**

**Node\* parent**

**};**

# MÔ TẢ THUẬT TOÁN

**Mã giả trong C/C++ thuật toán A\***

**Node result[size(Ma trận) \* size(Ma trận)];**

**<priority-queue> A**

**A.Thêm\_node(Start)**

**result[Start.value] = -1;**

**loop do{**

**if (A = rỗng)**

**then return Thất\_bại;**

**Node temp = Lấy\_node\_ưu tiên nhất(A);**

**result[temp.value] = temp.parent.value;**

**if (temp = Goal)**

**then return Lời\_giải;**

**Mở\_rộng(temp,A);**

**}**

**Mã giả mở rộng node**

**int check[size(Ma trận) \* size(Ma trận)] = {0}**

**int step\_X[] = {-1,-1,-1,0,1,1,1,0};**

**int step\_Y[] = {-1,0,1,1,1,0,-1,-1};**

**function Mở\_rộng(node, A){**

**Node temp;**

**int i = 0;**

**loop do{**

**if (A.x + step\_X[i] và A.y + step\_Y[i] thuộc ma trận && Ô đó có check = 0 && Ô đó không phải là chướng ngại vật){**

**temp.x = A.x + step\_X[i];**

**temp.y = A.y + step\_Y[i];**

**temp.value = A.x \* size(Ma trận) + A.y;**

**temp.heuristic = Euclid(temp,Goal);**

**temp.parent = A;**

**A.Thêm\_node(temp);**

**Đánh dấu check = 1 là đã mở;**

**}**

**i += 1;**

**}**

**}**

**Mã giả lời giải**

**function Lời\_Giải(Goal){**

**Node temp = Goal;**

**printf("(%d,%d)", Goal.x, Goal.y);**

**while (result[temp.parent.value] != -1){**

**printf(" <-- ");**

**printf("(%d,%d)", temp.parent.x, temp.parent.y);**

**temp = temp.parent;**

**}**

**}**

# CÁC BỘ TEST

**Bộ dữ liệu 1:**

Điểm start là (0,0)

Điểm goal là (6,6)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bộ dữ liệu 2:**

Điểm start là (4,1)

Điểm goal là (0,2)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |

**Bộ dữ liệu 3:**

Điểm start là (3,0)

Điểm goal là (0,0)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |

# MỨC ĐỘ HOÀN THÀNH VÀ ĐÁNH GIÁ

|  |  |  |  |
| --- | --- | --- | --- |
| **Công việc** | **Người thực hiện** | **Mục tiêu/ sản phẩm** | **Đánh giá (%)** |
| Viết báo cáo | Khải | File báo cáo | 100% |
| Vẽ sơ đồ hệ thống | Huy | Ảnh sơ đồ | 100% |
| Vẽ cấu trúc dữ liệu | Huy | Ảnh UML | 100% |
| Viết mô tả thuật toán | Khải | Văn bản trong phần báo cáo | 100% |
| Xây dựng các bộ test | Huy | Các file input và output tương ứng | 100% |
| Code thuật toán | Khải, Huy | Source code | 100% |