TRƯỜNG ĐẠI HỌC THỦY LỢI

**KHOA CÔNG NGHỆ THÔNG TIN**

A blue and white logo

Description automatically generated with low confidence

**BÀI TẬP LỚN**

HỌC PHẦN: TRÍ TUỆ NHÂN TẠO

**ĐỀ TÀI: TÌM ĐƯỜNG RA KHỎI MÊ CUNG**

Giáo viên hướng dẫn: Nguyễn Thị Kim Ngân

Sinh viên/nhóm sinh viên thực hiện:

1. Nguyễn Gia Tuấn - 2051063712

2. Bùi Xuân Hiếu - 1851061359

3. Hoàng Thị Phương Liên - 2051063679

**Hà Nội, năm 2021**

**Phần 1: Tổng quan**

1. ***Phương pháp***

**A** **Giải thuật tìm kiếm theo chiều rộng**

Giải thuật tìm kiếm theo chiều rộng (Breadth First Search – viết tắt là BFS) duyệt qua một đồ thị theo chiều rộng và sử dụng hàng đợi (queue) để ghi nhớ đỉnh liền kề để bắt đầu việc tìm kiếm khi không gặp được đỉnh liền kề trong bất kỳ vòng lặp nào.

Việc triển khai BFS tiêu chuẩn sẽ thực hiện chia mỗi đỉnh của biểu đồ thành một trong hai loại:

1. Đã được duyệt.
2. Không được duyệt.

Mục đích của thuật toán là đánh dấu mỗi đỉnh là đã duyệt trong khi tránh thực hiện quá trình duyệt theo vòng tròn.

Cách thuật toán hoạt động như sau:

1. Bắt đầu bằng cách đặt bất kỳ một trong các đỉnh của biểu đồ ở phía cuối của hàng đợi.
2. Lấy phần tử đầu tiên của hàng đợi và thêm nó vào danh sách đã truy cập.
3. Tạo danh sách các nút liền kề của đỉnh đó. Thêm những nút mà không có trong danh sách đã truy cập vào phía sau hàng đợi.
4. Tiếp tục lặp lại bước 2 và 3 cho đến khi hàng đợi trống rỗng.

**B** **Giải thuật tìm kiếm theo chiều sâu**

Giải thuật tìm kiếm theo chiều sâu (Depth First Search – viết tắt là DFS), còn được gọi là giải thuật tìm kiếm ưu tiên chiều sâu, là giải thuật duyệt hoặc tìm kiếm trên một cây hoặc một đồ thị và sử dụng stack (ngăn xếp) để ghi nhớ đỉnh liền kề để bắt đầu việc tìm kiếm khi không gặp được đỉnh liền kề trong bất kỳ vòng lặp nào.

Việc triển khai BFS tiêu chuẩn sẽ thực hiện chia mỗi đỉnh của biểu đồ thành một trong hai loại:

1. Đã được duyệt.
2. Không được duyệt.

Mục đích của thuật toán là đánh dấu mỗi đỉnh là đã duyệt trong khi tránh việc duyệt theo vòng tròn chu kỳ.

Thuật toán DFS hoạt động như sau:

1. Bắt đầu bằng cách đặt bất kỳ một trong các đỉnh của biểu đồ lên trên một ngăn xếp.
2. Lấy ra phần tử trên cùng của ngăn xếp và thêm nó vào danh sách đã truy cập.
3. Tạo danh sách các nút liền kề của đỉnh đó. Thêm những nút mà không có trong danh sách đã được duyệt vào đầu ngăn xếp.
4. Tiếp tục lặp lại các bước 2 và 3 cho đến khi ngăn xếp trống rỗng.

**Phần 2: Thực nghiệm (code)**

***1.Mô tả bài toán***

* **Tên bài toán**: Sử dụng phương pháp giải thuật tìm kiếm theo chiều rộng và tìm kiếm theo chiều sâu áp dụng vào bài toán tìm kiếm đường ra khỏi mê cung.

Coi như chúng ta có một mê cung cần được giải quyết.

Mục tiêu là di chuyển xung quanh mê cung để tìm được đường ra ( nghĩa là bạn sẽ được cung cấp 1 vị trí bắt đầu tại lối vào mê cung và quá trình chỉ kết thúc khi bạn có mặt tại cánh cửa lối ra mê cung hay còn gọi là điểm đích)

**Input**:

* + Một mê cung dạng lưới kích thước 12x12
  + Điểm bắt đầu
  + Điểm kết thúc
* **Ouput:** 
  + Tổng quãng đường đã di chuyển
  + Quãng đường ngắn nhất để đi ra khỏi mê cung
  + Thời gian thực hiện ứng với từng phương pháp
* Method (phương pháp thực hiện):

+ Mô tả cách xây dựng trạng thái.

Hình thành bài toán bằng việc xác định 4 tham số:

* + Initial State
  + Successor
  + Goal Test
  + Path Cost

+ Các quy luật chuyển trạng thái:

Các hành động khả dụng là di chuyển trái, phải, lên xuống.

+ Mô tả trạng thái đầu, trạng thái đích.

Điểm bắt đầu được đặt tại vị trí 5x12 trong mê cung

Điểm kết thúc được đặt tại vị trí 11x1 trong mê cung

+ Mô tả cách thực hiện để tìm ra đường đi từ trạng thái đầu đến trạng thái đích.

* Sử dụng phương pháp tìm kiếm theo chiều rộng

***2. Đánh giá kết quả tìm được:***

- Số các bước chuyển trạng thái từ trạng thái đầu đến trạng thái đích của mỗi phương pháp duyệt. So sánh kết quả của 2 phương pháp.

- So sánh thời gian thực hiện của 2 phương pháp.

**Kết luận**

* Tóm lược các nội dung chính mà bài tập lớn làm được

**Tài liệu tham khảo**

* Các tài liệu được tham khảo trong báo cáo