Ảnh có chứa Hình chữ nhật, khung, khung ảnh

Mô tả được tạo tự động**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM THÀNH PHỐ HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**Ảnh có chứa biểu tượng, Phông chữ, văn bản, Đồ họa

Mô tả được tạo tự động**

**BÁO CÁO ĐỒ ÁN**

**LÝ THUYẾT ĐỒ THỊ VÀ ỨNG DỤNG**

**Đề tài: Minh họa trực quan và demo ứng dụng với thuật toán lý thuyết đồ thị Prim và Dijkstra**

**Giảng viên hướng dẫn: ThS. Trần Quang Huy**

**TS. Nguyễn Viết Hưng**

**Mã học phần: COMP170101**

**Nhóm sinh viên thực hiện:**

Họ và tên: Phan Vũ Tuấn Anh - MSSV: 47.01.104.045

Họ và tên: Quách Tuấn Anh - MSSV: 48.01.104.005

Họ và tên: Nguyễn Ngọc Loan - MSSV: 48.01.104.079

***Thành phố Hồ Chí Minh, ngày 16 tháng 12 năm 2023***

LỜI CÁM ƠN

Lời đầu tiên, chúng em xin trân trọng cảm ơn và bày tỏ lòng biết ơn sâu sắc đến thầy **Nguyễn Viết Hưng** và thầy **Trần Quang Huy** Giảng viên khoa Công nghệ thông tin trường Đại học Sư Phạm Thành Phố Hồ Chí Minh, giáo viên hướng dẫn môn Lý thuyết đồ thị và ứng dụng cho tụi em thật nhiệt tình.

Chúng em xin chân thành cảm ơn các thầy cô giáo đang giảng dạy tại khoa CNTT trường Đại học Sư Phạm Thành phố Hồ Chí Minh đã nhiệt tình ủng hộ, hướng dẫn và truyền lại cho chúng em rất nhiều kiến thức quý báu!.

Cuối cùng, Nhóm chúng mình xin chân thành cảm ơn bạn bè đã hỗ trợ và trao đổi đóng góp ý kiến để nhóm chúng mình hoàn thành được đề tài này.

Xin chân thành cảm ơn.

**Nhóm sinh viên thực hiện Báo cáo**

**Quách Tuấn Anh**

**Phan Vũ Tuấn Anh**

**Nguyễn Ngọc Loan**

**Mục Lục**

PHÂN CÔNG VÀ ĐÁNH GIÁ MỨC ĐỘ HOÀN THÀNH

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **HỌ VÀ TÊN -**  **MÃ SỐ SINH VIÊN** | **PHÂN CÔNG NHIỆM VỤ** | **ĐÁNH GIÁ MỨC ĐỘ HOÀN THÀNH** |
| 1 | Phan Vũ Tuấn Anh –  47.01.104.045 | Tìm hiểu code, viết code chính, thuyết trình và demo sản phẩm | 100% |
| 2 | Quách Tuấn Anh -  48.01.104.005 | Tìm hiểu code, viết báo cáo word | 100% |
| 3 | Nguyễn Ngọc Loan –  48.01.104.079 | Tìm hiểu code, viết báo cáo power point | 100% |

ĐÁNH GIÁ, NHẬN XÉT CỦA GIẢNG VIÊN

- Điểm số:

- Điểm chữ:

*Thành phố Hồ Chí Minh, ngày 16 tháng 12 năm 2023***GIẢNG VIÊN**

**ThS. Trần Quang Huy**

**TS. Nguyễn Viết Hưng**

DANH MỤC HÌNH VẼ VÀ ĐỒ THỊ

PHẦN NỘI DUNG

CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI

1.Lý do chọn đề tài

Chọn đề tài Minh họa trực quan và demo ứng dụng với thuật toán lý thuyết đồ thị Prim và Dijkstra là một quyết định tốt vì nó kết hợp hai thuật toán quan trọng trong lĩnh vực lý thuyết đồ thị và có thể mang lại nhiều lợi ích:

a. Độ phổ biến của thuật toán: Cả hai thuật toán Prim và Dijkstra đều rất phổ biến và được sử dụng rộng rãi trong lĩnh vực khoa học máy tính và công nghệ thông tin. Hiểu rõ về cách chúng hoạt động và cách thức triển khai sẽ giúp bạn hiểu sâu hơn về các thuật toán quan trọng này.

b. Áp dụng trong thực tế: Cả Prim và Dijkstra được áp dụng rộng rãi trong thực tế. Thuật toán Prim được sử dụng trong việc tìm cây khung nhỏ nhất của đồ thị, trong khi Dijkstra được sử dụng để tìm đường đi ngắn nhất từ một đỉnh đến tất cả các đỉnh còn lại trong đồ thị. Việc minh họa và demo ứng dụng sẽ giúp người khác hiểu rõ cách áp dụng chúng trong các vấn đề thực tế như định tuyến mạng, quản lý tài nguyên, hoặc lập lịch.

c. Trực quan hóa và giảng dạy: Minh họa trực quan và demo ứng dụng sẽ giúp bạn trình bày thuật toán một cách dễ hiểu hơn cho người khác. Việc thực hiện các ví dụ minh họa có thể bao gồm việc sử dụng đồ họa, animation hoặc giao diện người dùng tương tác để giải thích cách thuật toán hoạt động.

d. Nền tảng cho nghiên cứu tiếp theo: Hiểu rõ về cách thức hoạt động của Prim và Dijkstra có thể là cơ sở cho các nghiên cứu tiếp theo trong lĩnh vực lý thuyết đồ thị hoặc phát triển các thuật toán mới dựa trên chúng.

e. Thách thức và cơ hội: Thực hiện demo và minh họa có thể là một thách thức, nhưng cũng đồng thời là cơ hội để bạn phát triển kỹ năng lập trình, hiểu sâu hơn về cách làm việc của các thuật toán này và cách tương tác với người dùng.

Nhớ rằng, việc chọn đề tài này đòi hỏi sự hiểu biết sâu rộng về cả hai thuật toán. Đồng thời, việc thực hiện demo và minh họa cần có sự tập trung cao và kỹ năng lập trình tốt để có thể thể hiện được cách thức hoạt động của chúng một cách trực quan và dễ hiểu.

2. Mục tiêu phát triển

Mục tiêu phát triển khi nghiên cứu và thực hiện demo về thuật toán Prim và Dijkstra bao gồm:

a. Hiểu sâu hơn về lý thuyết đồ thị: Điều này bao gồm việc hiểu rõ cách thức hoạt động của đồ thị, các loại đồ thị khác nhau và cách mà thuật toán Prim và Dijkstra được áp dụng trong các tình huống khác nhau

b. Nâng cao kỹ năng lập trình: Việc thực hiện demo yêu cầu kỹ năng lập trình tốt. Bạn có thể phát triển kỹ năng này thông qua việc xây dựng các ứng dụng hoặc demo tương tác trên nền tảng máy tính.

c.Tạo ra một sản phẩm demo chất lượng cao: Mục tiêu là tạo ra một demo hoặc ứng dụng trực quan và dễ sử dụng, giúp người dùng hiểu rõ về cách hoạt động của các thuật toán này một cách thực tế.

d. Giao tiếp và giảng dạy: Phát triển kỹ năng giao tiếp và khả năng giải thích một cách dễ hiểu về các khái niệm phức tạp như thuật toán lý thuyết đồ thị cho người khác, có thể là sinh viên, đồng nghiệp hoặc cả những người không có kiến thức chuyên sâu về lĩnh vực này.

e. Tạo cơ sở cho nghiên cứu và ứng dụng thực tế: Hiểu biết sâu rộng về các thuật toán này có thể tạo ra cơ hội cho việc thực hiện các dự án nghiên cứu hoặc áp dụng trong thực tế, như trong lĩnh vực mạng, tối ưu hóa, hoặc công nghệ thông tin.

f. Đóng góp vào cộng đồng học thuật: Việc chia sẻ kiến thức thông qua demo và giải thích sẽ có thể giúp cộng đồng học thuật hiểu rõ hơn về các thuật toán này và tạo ra sự chia sẻ kiến thức hữu ích..

3. Đối tượng

Tất cả mọi người

4. Mô tả thuật toán

a. Thuật Toán Prim:

Thuật toán Prim là một thuật toán sử dụng trong lý thuyết đồ thị để tìm cây khung nhỏ nhất của một đồ thị vô hướng, liên thông và có trọng số. Được phát triển bởi nhà toán học Robert C. Prim vào năm 1957 và sau đó được nâng cấp bởi nhà toán học Edsger W. Dijkstra và nhà toán học Joseph Kruskal.

Mô tả thuật toán:

-Khởi tạo: Chọn một đỉnh bất kỳ làm đỉnh bắt đầu của cây khung.

-Tạo tập hợp các cạnh: Duyệt qua tất cả các cạnh kề với đỉnh ban đầu và thêm chúng vào một tập hợp các cạnh có thể chọn.

-Lặp cho đến khi có cây khung hoàn chỉnh:

+ Chọn cạnh có trọng số nhỏ nhất từ tập hợp các cạnh có thể chọn sao cho đỉnh kết thúc của cạnh không thuộc cây khung và đưa đỉnh đó vào cây khung.

+Cập nhật tập hợp các cạnh có thể chọn bằng cách thêm các cạnh kề với đỉnh mới thêm vào cây khung.

-Kết thúc khi cây khung hoàn chỉnh: Thuật toán dừng lại khi tất cả các đỉnh trong đồ thị được thêm vào cây khung, tạo thành một cây khung nhỏ nhất với trọng số tổng của các cạnh là nhỏ nhất.

Đặc điểm quan trọng của thuật toán Prim:

- Tính đúng đắn: Đảm bảo tìm ra cây khung nhỏ nhất của đồ thị.

- Hiệu suất cao: Có thể sử dụng các cấu trúc dữ liệu như hàng đợi ưu tiên để tối ưu hóa việc chọn cạnh có trọng số nhỏ nhất.

- Thời gian chạy: Thời gian chạy của thuật toán Prim phụ thuộc vào cách thức lưu trữ đồ thị, nhưng trong trường hợp tốt nhất, nó có thể chạy với độ phức tạp O(E log V), với E là số cạnh và V là số đỉnh trong đồ thị.

Thuật toán Prim cung cấp một cách hiệu quả để tìm cây khung nhỏ nhất trong các đồ thị có trọng số, và nó được áp dụng rộng rãi trong nhiều lĩnh vực như mạng máy tính, truyền thông, và tối ưu hóa đường đi.

b. Thuật Toán Dijkstra

Thuật toán Dijkstra được sử dụng để tìm đường đi ngắn nhất từ một đỉnh đến tất cả các đỉnh còn lại trong đồ thị có trọng số không âm. Được phát triển bởi nhà toán học Edsger W. Dijkstra vào năm 1956.

Mô tả thuật toán:

-Khởi tạo: Đặt đỉnh ban đầu có đường đi từ chính nó bằng 0 và các đỉnh khác bằng vô cùng (hoặc một giá trị lớn đủ lớn để không bị ảnh hưởng đến kết quả).

-Lặp cho đến khi tất cả các đỉnh được xét:

+Chọn đỉnh với đường đi nhỏ nhất từ các đỉnh chưa được xét.

+Cập nhật đường đi của tất cả các đỉnh kề với đỉnh vừa chọn nếu có đường đi mới ngắn hơn đường đi hiện tại của chúng. Điều này dựa trên việc cộng trọng số cạnh với đường đi từ đỉnh đã chọn đến đỉnh đó.

-Kết thúc khi tất cả các đỉnh được xét:

+Thuật toán dừng khi tất cả các đỉnh đã được xét và có đường đi ngắn nhất từ đỉnh ban đầu đến tất cả các đỉnh khác trong đồ thị.

Đặc điểm quan trọng của thuật toán Dijkstra:

-Đường đi ngắn nhất: Đảm bảo tìm ra đường đi ngắn nhất từ một đỉnh đến tất cả các đỉnh còn lại trong đồ thị.

- Giả thiết trọng số không âm:Thuật toán này hoạt động hiệu quả khi các trọng số trên cạnh đồ thị không âm.

- Tối ưu hóa: Sử dụng hàng đợi ưu tiên để lựa chọn đỉnh với đường đi ngắn nhất để xét, giúp cải thiện hiệu suất của thuật toán.

Thuật toán Dijkstra rất hữu ích trong việc tìm đường đi ngắn nhất trong mạng lưới giao thông, mạng máy tính, quản lý tài nguyên, và nhiều ứng dụng khác. Tuy nhiên, điều kiện trọng số không âm là một giả thiết quan trọng, và nó không hoạt động chính xác khi có trọng số âm trong đồ thị.

PHẦN THỰC HÀNH

CHƯƠNG 2: CHI TIẾT VÀ DEMO SẢN PHẨM