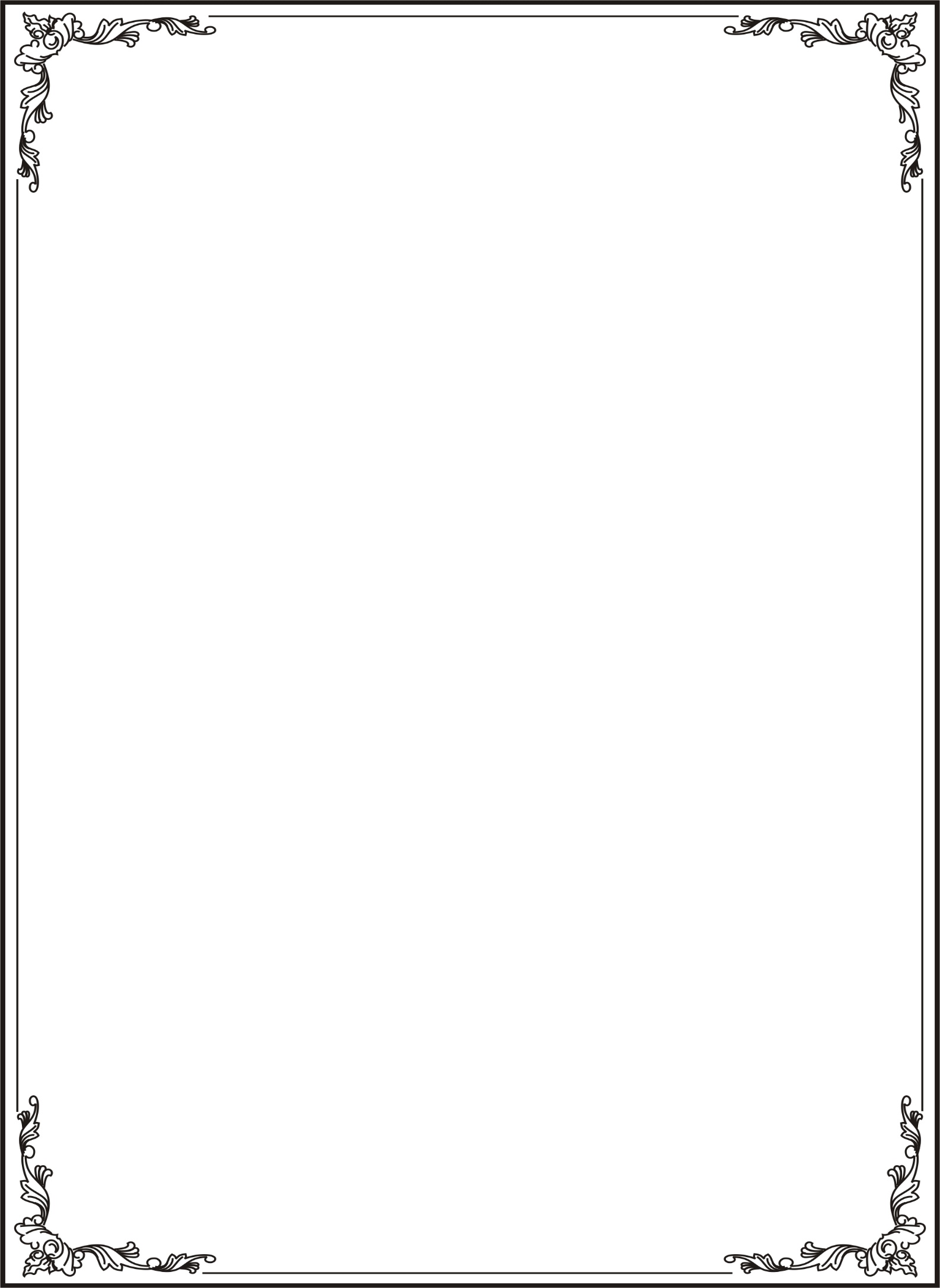
**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM**

**KHOA CÔNG NGHỆ THÔNG TIN**



BÁO CÁO ĐỒ ÁN CUỐI KÌ

**ĐỀ TÀI: A Generalization of Dijkstra’s Algorithm**

Giảng viên hướng dẫn: TS. Huỳnh Xuân Phụng

Danh sách sinh viên thực hiện:

Đặng Thanh Bình – 21110380

Nguyễn Thiện Luân – 21110538

Ngô Minh Lợi – 21110898

TP. Hồ Chí Minh, tháng 12, năm 2022

Nhận xét của giảng viên:

………………………………………………………………………………………….

………………………………………………………………………………………….

………………………………………………………………………………………….

………………………………………………………………………………………….

………………………………………………………………………………………….

………………………………………………………………………………………….

………………………………………………………………………………………….

………………………………………………………………………………………….

………………………………………………………………………………………….

………………………………………………………………………………………….

………………………………………………………………………………………….

………………………………………………………………………………………….

………………………………………………………………………………………….

………………………………………………………………………………………….

………………………………………………………………………………………….

Kí tên:

**MỤC LỤC**

[Chương 1: Phân tích các đặc điểm của thuật toán Dijkstra 1](#_Toc123134756)

[1.1. Input 1](#_Toc123134757)

[1.2. Output 1](#_Toc123134758)

[1.3. Các thuật ngữ của mindmap 1](#_Toc123134759)

[Chương 2: Thuật toán Dijkstra 4](#_Toc123134760)

[2.1. Lưu đồ thuật toán 4](#_Toc123134761)

[2.2. Các bước thực hiện 5](#_Toc123134762)

[2.3. Ứng dụng của thuật toán 5](#_Toc123134763)

[2.3.1. Ứng dụng tìm đường đi ngắn nhất trên bản đồ 5](#_Toc123134764)

[2.3.2. Ứng dụng trong mạng xã hội 6](#_Toc123134765)

[2.3.3. Ứng dụng trong hệ thống thông tin di động điện tử 6](#_Toc123134766)

[2.3.4. Ứng dụng trong kỹ thuật của ngành hàng không vũ trụ 6](#_Toc123134767)

[Chương 3: Source code của thuật toán: 8](#_Toc123134768)

[3.1. Thuật toán 8](#_Toc123134769)

[3.2. Mô phỏng thuật toán 15](#_Toc123134770)

[3.2.1. Tìm đường đi ngắn nhất 15](#_Toc123134774)

[3.2.2. Ứng dụng trong mạng xã hội 16](#_Toc123134775)

[Poster của thuật toán Dijkstra: 17](#_Toc123134776)

**Thuật toán Dijkstra**, mang tên của nhà khoa học máy tính người Hà Lan [Edsger Dijkstra](https://vi.m.wikipedia.org/wiki/Edsger_Dijkstra) vào năm 1956 và ấn bản năm 1959, là một [thuật toán](https://vi.m.wikipedia.org/wiki/Thu%E1%BA%ADt_to%C3%A1n) giải quyết [bài toán đường đi ngắn nhất](https://vi.m.wikipedia.org/wiki/B%C3%A0i_to%C3%A1n_%C4%91%C6%B0%E1%BB%9Dng_%C4%91i_ng%E1%BA%AFn_nh%E1%BA%A5t) từ một đỉnh đến các đỉnh còn lại của [đồ thị có hướng](https://vi.m.wikipedia.org/wiki/%C4%90%E1%BB%93_th%E1%BB%8B_(l%C3%BD_thuy%E1%BA%BFt_%C4%91%E1%BB%93_th%E1%BB%8B)#%C4%91%E1%BB%93_th%E1%BB%8B_c%C3%B3_h%C6%B0%E1%BB%9Bng) không có [cạnh](https://vi.m.wikipedia.org/wiki/%C4%90%E1%BB%93_th%E1%BB%8B_(l%C3%BD_thuy%E1%BA%BFt_%C4%91%E1%BB%93_th%E1%BB%8B)) mang trọng số không âm. Thuật toán thường được sử dụng trong [định tuyến](https://vi.m.wikipedia.org/wiki/%C4%90%E1%BB%8Bnh_tuy%E1%BA%BFn) với một [chương trình con](https://vi.m.wikipedia.org/wiki/Ch%C6%B0%C6%A1ng_tr%C3%ACnh_con) trong các thuật toán đồ thị hay trong công nghệ [Hệ thống định vị toàn cầu](https://vi.m.wikipedia.org/wiki/H%E1%BB%87_th%E1%BB%91ng_%C4%91%E1%BB%8Bnh_v%E1%BB%8B_to%C3%A0n_c%E1%BA%A7u) ([GPS](https://vi.m.wikipedia.org/wiki/GPS)).

**Bài toán:**

Cho một đồ thị có hướng G = (V, E), một hàm trọng số w: E → [0, ∞) và một đỉnh nguồn s. Cần tính toán được đường đi ngắn nhất từ đỉnh nguồn s đến mỗi đỉnh của đồ thị.

Ví dụ: Chúng ta dùng các đỉnh của đồ thị để mô hình các thành phố và các cạnh để mô hình các đường nối giữa chúng. Khi đó trọng số các cạnh có thể xem như độ dài của các con đường (và do đó là không âm). Chúng ta cần vận chuyển từ thành phố s đến thành phố t. Thuật toán Dijkstra sẽ giúp chỉ ra đường đi ngắn nhất chúng ta có thể đi.

Trọng số không âm của các cạnh của đồ thị mang tính tổng quát hơn khoảng cách hình học giữa hai đỉnh đầu mút của chúng. Ví dụ, với 3 đỉnh A, B, C đường đi A-B-C có thể ngắn hơn so với đường đi trực tiếp A-C.

**Giới thiệu thuật toán:**

Xét đồ thị G = (X, E) với các cạnh có trọng số không âm. - Dữ liệu nhập cho thuật toán là ma trận trọng số L (với qui ước L(h, k) = +∞ nếu không có cạnh nối từ đỉnh h đến đỉnh k) và hai đỉnh x,y cho trước.

Dữ liệu xuất là đường đi ngắn nhất từ x đến y

Chương 1: Phân tích các đặc điểm của thuật toán Dijkstra

* 1. Input

- Nhập vào n đỉnh.

- Nhập lần lượt các cặp đỉnh có cạnh kề và trị số cạnh đấy nếu có.

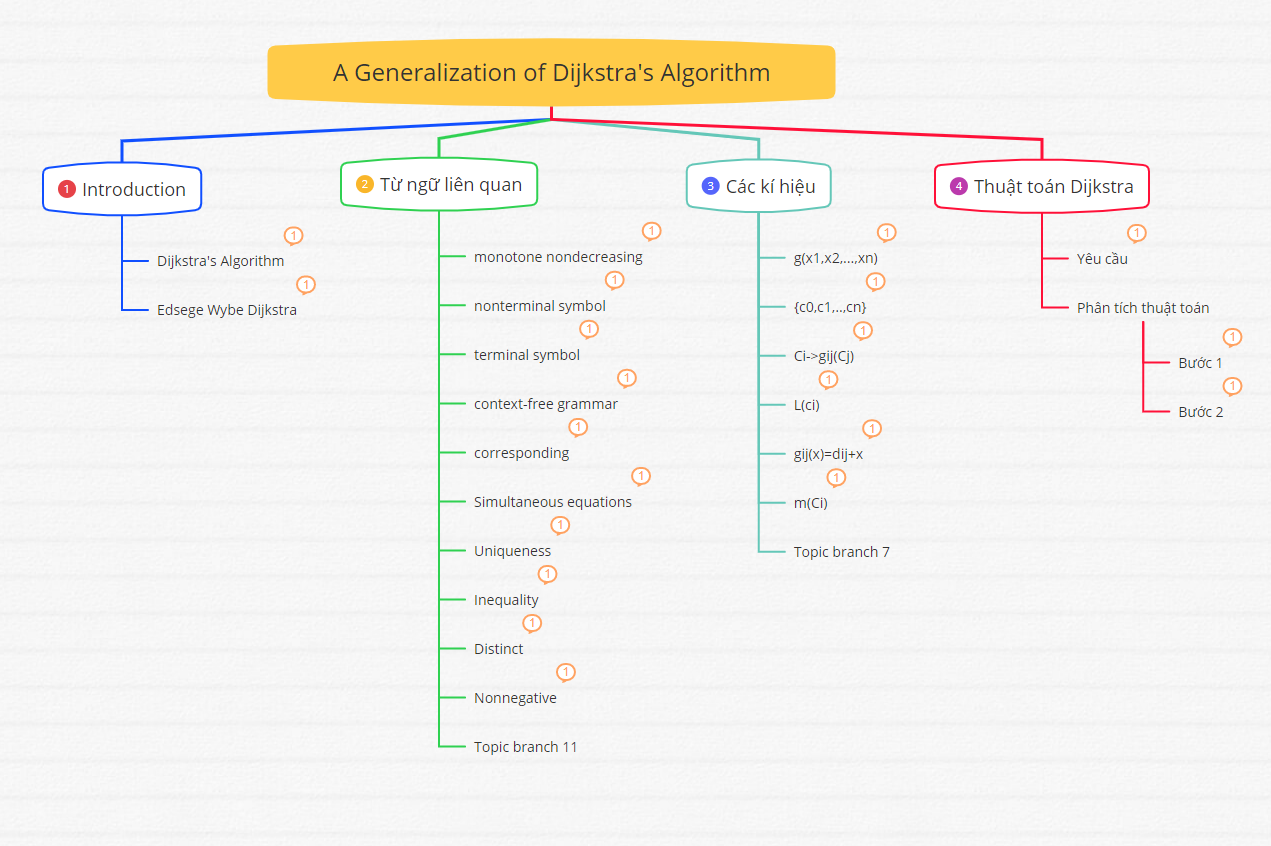
- Nhập đỉnh bắt đầu và kết thúc của đường đi muốn tìm.

* 1. Output

- Xuất ra độ dài ngắn nhất của đường đi muốn tìm.

- Xuất ra đường đi (thứ tự các đỉnh mà đường đi đi qua).

* 1. Các thuật ngữ của mindmap



* Introduction

Dijkstra’s Algorithm: Là thuật toán tìm đường đi ngắn nhất trong đồ thị có trọng số không âm

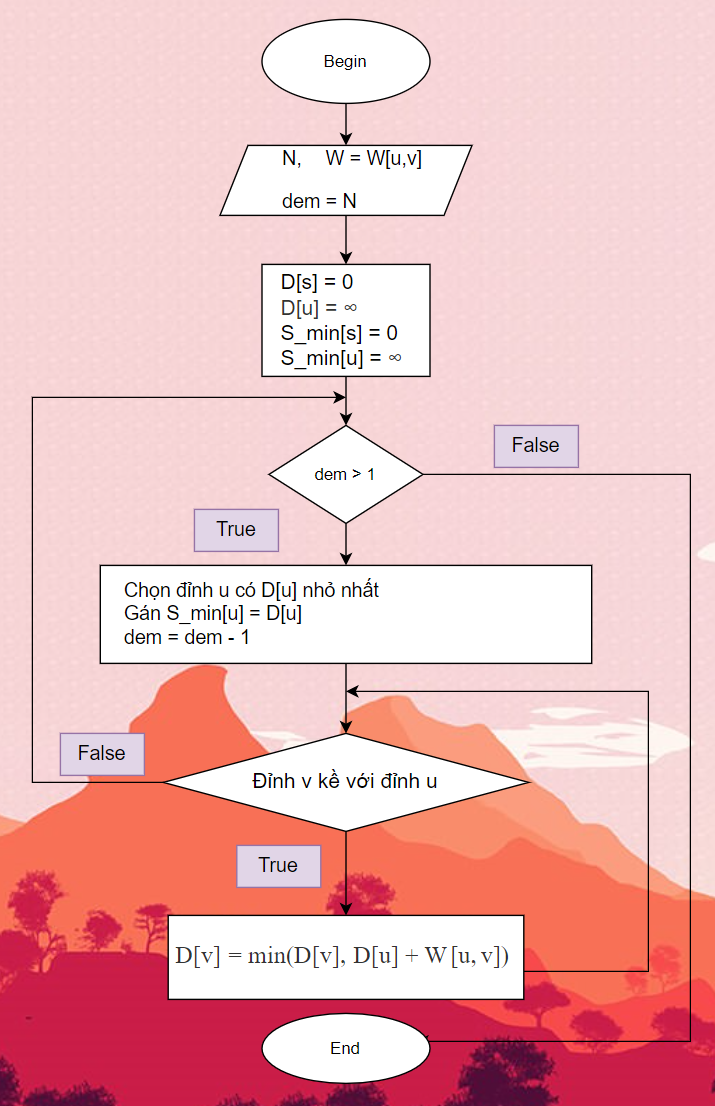
Edsger Wybe Dijkstra: nhà toán học Hà Lan tìm ra được thuật toán Dijkstra

* Từ ngữ liên quan
* general form: khuôn mẫu chung
* superior: vượt trội, siêu việt
* superior function: hàm siêu việt
* parenthese: dấu ngoặc đơn
* monotone nondecreasing: hàm đơn điệu không giảm
* nonterminal symbol: không là kí hiệu kết thúc còn được gọi là biến
* terminal symbol: là kí hiệu kết thúc
* context-free grammar: ngôn ngữ tự nhiên
* corresponding: tương ứng
* Simultaneous equations: phương trình đồng thời
* Inequality: bất phương trình
* Distinct: riêng biệt
* Nonnegative: không âm
* Uniqueness: tính duy nhất
* Emptiness testing: kiểm tra tính rỗng
* Optimum binary search tree: tối ưu hóa thuật toán cây nhị phân
* Các kí hiệu
* →\*: có thể dẫn đến kết quả cuối cùng là
* Ci->gij(Cj): từ điểm Ci tới Cj sẽ nối với nhau qua con đường gij
* C0 → 0: mốc tại điểm xuất phát sẽ có khoảng cách là 0
* g(x1, x2,..,xn): là hàm ánh xạ từ Rk+ to R+
* {c0, c1,..,cn}: tập hợp n+1 điểm đánh dấu đầu cuối
* L(ci): được coi như độ dài tương ứng từ điểm bắt đầu tới ci
* gij(x)=dij+x: dij là độ dài con đường; x là khoảng cách từ điểm xuất phát tới x
* m(Ci): là khoảng cách của đường đi ngắn nhất từ điểm xuất phát tới Ci

Chương 2: Thuật toán Dijkstra

1. 1. Lưu đồ thuật toán

Link truy câp: [Lastest Lưu đồ Dijkstra.drawio - diagrams.net](https://app.diagrams.net/#G1SABN4LTbCP6QI-VG-PQfFL65wbLbXyda)



* 1. Các bước thực hiện

Bước 1: Kiểm tra xem giá trị biến đến (int dem) có lớn hơn 1 không. Nếu không (dem = 1, tức là khi đó tất cả các điểm trên đồ thị đều đã được xử lí) thì kết thúc thuật toán. Thực hiện Bước 4.

Bước 2: Tìm trong tất cả đỉnh u trừ đỉnh gốc, đỉnh có giá trị D[u] nhỏ nhất giả sử nó là đỉnh u. Gán giá trị S\_min[u] = D[u]. Và giảm giá trị biến đếm 1 đơn vị (dem = dem- 1)

Bước 3: Xét trong tất cả các đỉnh v có đường đi gần kề với u, giả sử là đỉnh v thì ta thực hiện:

(nếu D[v] > (D[u] + W[u, v])

gán lại: D[v] = D[u] + W[u, v])

Quay lại Bước 1.

Bước 4: Xuất các giá trị S\_min[u].

* 1. Ứng dụng của thuật toán

Thuật toán Dijkstra có công năng chính của nó là thay thế con người tìm đường đi ngắn nhất mà chúng ta không thể tính toán bằng bộ não, ví dụ điển hình có thể là các app Google Map của Mỹ hay Baidu Map của Trung Quốc cũng một phần sử dụng thuật toán này.

Ứng dụng thực tiễn của thuật toán Dijkstra trong đời sống hiện nay:

1. * 1. Ứng dụng tìm đường đi ngắn nhất trên bản đồ

Theo đó các ứng dụng tìm kiếm đường đi và chỉ đường hiện nay đều sẽ hiện nhiều lựa chọn với các trị số thời gian để bạn lựa chọn ra con đường đi ngắn nhất từ xuất phát điểm đến dựa trên những hiển thị và các yếu tố từ vệ tinh, từ đó áp dụng thuật toán Dijkstra để hiển thị đường đi

* + 1. Ứng dụng trong mạng xã hội

Các doanh nghiệp kinh doanh nhỏ lẻ hay các trang mạng xã hội có hướng dẫn đường đi cho người theo dõi cũng áp dụng thuật toán Dijkstra để cnhungsd dường đi của doanh nghiệp lên mạng xã hội. Qua đó, người dùng chỉ cần truy cập vào trang facebook của doanh nghiệp, sử dụng chức năng chỉ đường là sẽ tự động được tính toán và dẫn ra con đường ngắn nhất.

* + 1. Ứng dụng trong hệ thống thông tin di động điện tử

Ngoài việc tìm đường đi thực tế, một số hệ thống thông tin di đọng còn ứng dụng thuật toán này để có thể truyền tải thông tin nhanh hơn khi có kết nối nội bộ giữa các đỉnh, các đỉnh này có thể là GPS hay Airdrop, miễn là có thể kết nối thì thuật toán sẽ tìm đựơc đường đi nhanh nhất để truyền tải thông tin bạn mong muốn.

Bên cạnh đó, việc sử dụng internet cũng là điều kiện để các hacker sử dụng dấu vết của bạn, kết nối các đinh và truy tìm ra những thông tin được kết nối cũng như đường đi chính xác và ngắn nhất đến địa điểm mà bạn đang truy cập mạng.

* + 1. Ứng dụng trong kỹ thuật của ngành hàng không vũ trụ

Tương tự như hệ thống giao thông vận tải mặt đất, thuật toán Dijkstra cực kì hữu dụng khi các phi công phải dựa trên bản đồ hiển thị trong quá trình lái máy bay được tích hợp thông qua thuật toán, tránh việc tìm đường dựa trên cảm quan gây ra những sai sót nghiêm trọng đến tính mạng cũng như các hệ lụy nặng nề khác.

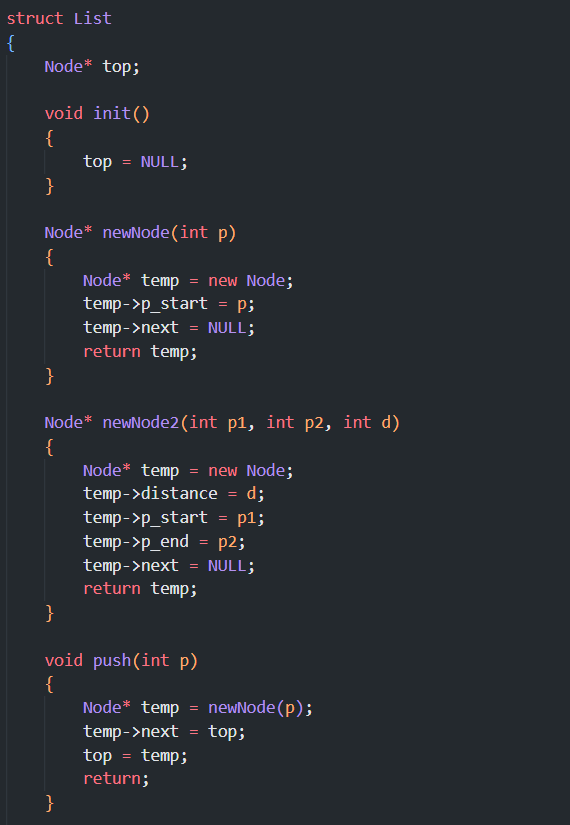
Tính chất của ngành hàng không là phải bay theo quỹ đạo được định sẵn bởi thuật toán nếu bạn cố ý bay chệch đường bay được định sẵn, thuật toán sẽ trở nên lộn xộn và vượt ra ngoài tầm kiểm soát.

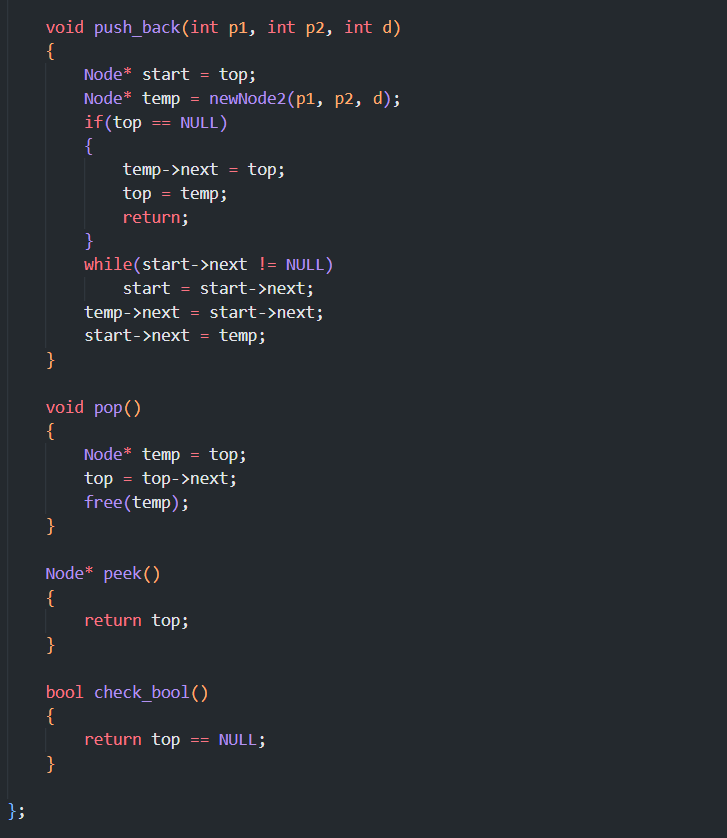
Chương 3: Source code của thuật toán:

1. 1. Thuật toán

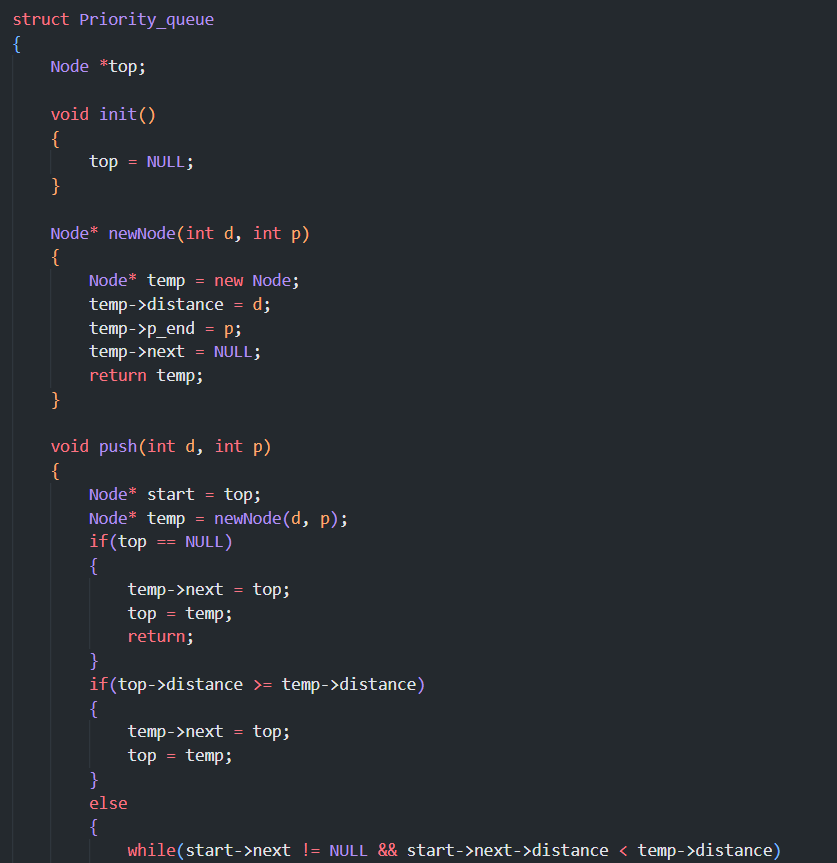
Để cài đặt Thuật toán, nhóm chúng em sử dụng 2 cấu trúc: List & Priority Queue

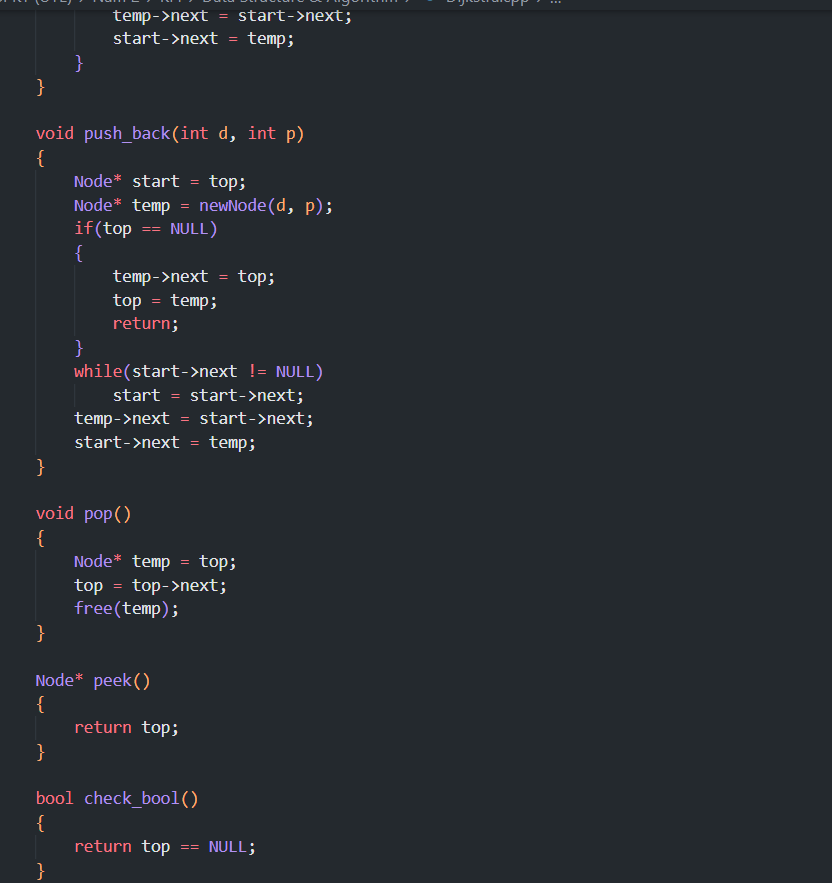
* + - * Cấu trúc List



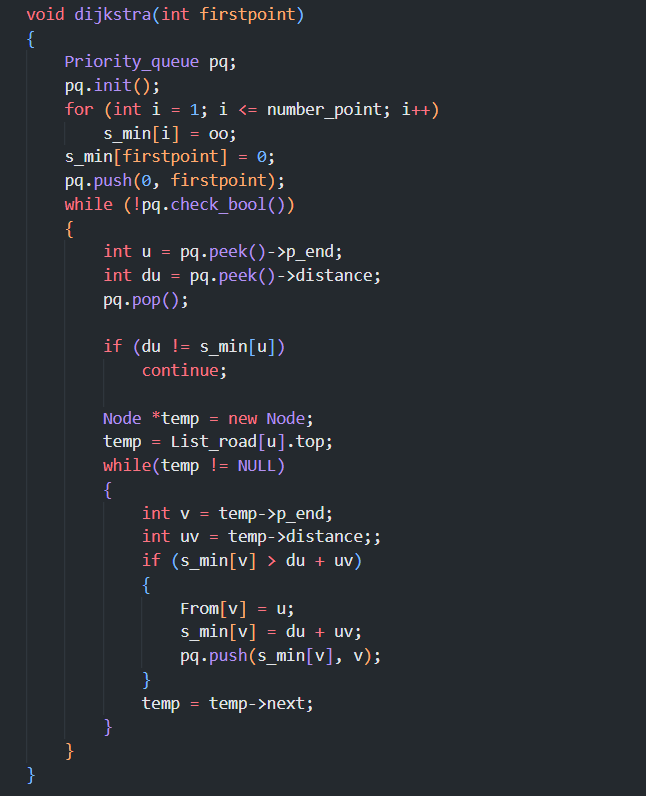


* List dùng để lưu trữ các đỉnh mà đường đi đi qua, dùng cho mục đích truy vết.
  + - * Cấu trúc Priority\_Queue

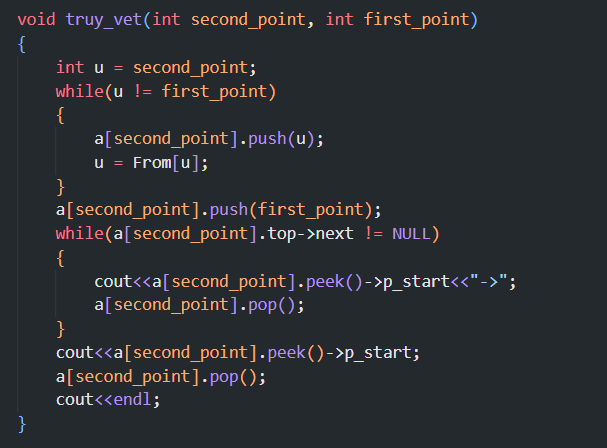




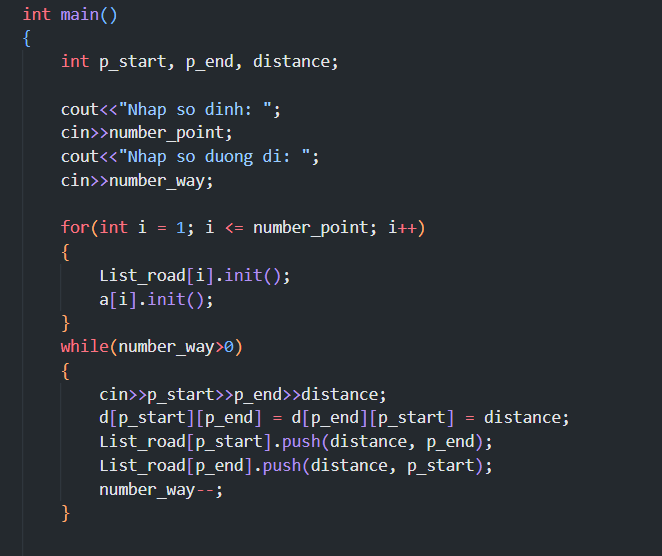
* Priority\_queue: Là cấu trúc hàng đợi ưu tiên, đưa các phần tử vào có điều kiện. Ở đây nhóm em sử dụng điều kiện sắp xếp theo thứ tự đưa vào từ bé đến lớn. Mục đích để có thể dễ dàng lấy ra phần tử nhỏ nhất, tối ưu thời gian chạy chương trình và độ phức tạp thuật toán.
* Priority\_queue: Sử dụng với các mục đích lưu trữ tập hợp các đường đi tồn tại trong đồ thị để thực hiện thuật toán Dijkstra. Và dùng để lưu tập hợp các đường đi giữa các điểm kề nhau để có thể xử lí thuật toán.
  + - * Thuật toán Dijkstra:

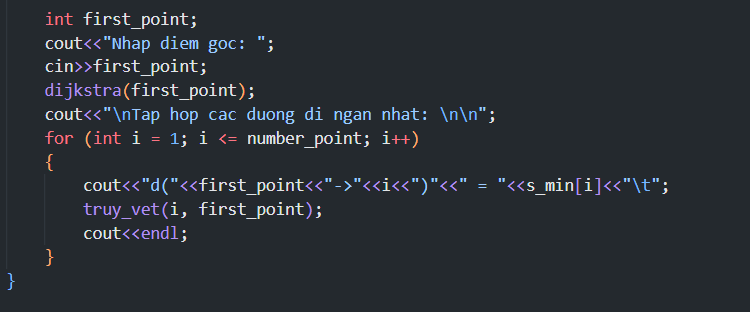


* + - * Hàm Truy vết:



* + - * Hàm main:





* 1. Mô phỏng thuật toán

1. 2. 1. Tìm đường đi ngắn nhất

A screenshot of a video game

Description automatically generated with medium confidence

* + 1. Ứng dụng trong mạng xã hội

A picture containing timeline

Description automatically generated

Poster của thuật toán Dijkstra:

Link truy cập: [diagrams.net](https://app.diagrams.net/#G1Rwb_zGPmZP5NbfMzQT2CrnWzd-505gMx)

