## Bài thực hành 13:

## CÀI ĐẶT BÀI TOÁN TÌM ĐƯỜNG ĐI NGẮN NHẤT

## Mô tả bài toán

### Đầu vào

Đồ thị: Đầu vào yêu cầu một đồ thị vô hướng hoặc có hướng, có thể được biểu diễn bằng ma trận kề hoặc danh sách kề. Trong đồ thị này, các đỉnh biểu diễn các điểm trong mạng hoặc hệ thống, và các cạnh biểu diễn mối quan hệ giữa các điểm đó. Mỗi cạnh có thể có một trọng số dương, đại diện cho chi phí, khoảng cách hoặc thời gian đi qua cạnh đó.

Đỉnh nguồn: Đầu vào yêu cầu chỉ định đỉnh nguồn (điểm xuất phát) trong đồ thị. Thuật toán Dijkstra sẽ tìm đường đi ngắn nhất từ đỉnh nguồn đến tất cả các đỉnh còn lại trong đồ thị.

### Đầu ra

Khoảng cách từ đỉnh nguồn đến tất cả các đỉnh còn lại trong đồ thị. Đầu ra thường được lưu trữ trong một mảng, trong đó vị trí của mỗi phần tử trong mảng tương ứng với chỉ số của đỉnh. Giá trị tại mỗi vị trí đại diện cho khoảng cách từ đỉnh nguồn đến đỉnh tương ứng.

### Cách xử lý

Cách xử lý của chương trình này dựa trên thuật toán Dijkstra để tìm đường đi ngắn nhất từ đỉnh nguồn đến các đỉnh còn lại trong đồ thị có trọng số dương. Dưới đây là các bước chính trong quá trình xử lý của chương trình:

Khởi tạo khoảng cách ban đầu: Trước khi bắt đầu thuật toán Dijkstra, tất cả các đỉnh được khởi tạo với khoảng cách vô cùng lớn (đại diện cho khoảng cách không xác định). Khoảng cách của đỉnh nguồn được đặt là 0, vì khoảng cách từ đỉnh nguồn đến chính nó là 0.

Sử dụng hàng đợi ưu tiên: Chương trình sử dụng một hàng đợi ưu tiên để lưu trữ các đỉnh theo thứ tự tăng dần của khoảng cách. Đỉnh nguồn được thêm vào hàng đợi ưu tiên ban đầu. Trong quá trình thực hiện thuật toán, đỉnh có khoảng cách nhỏ nhất sẽ luôn được lấy ra từ hàng đợi ưu tiên.

Duyệt qua các đỉnh và cập nhật khoảng cách: Trong vòng lặp chính, chương trình lấy đỉnh có khoảng cách nhỏ nhất từ hàng đợi ưu tiên. Sau đó, nó duyệt qua tất cả các đỉnh kề của đỉnh hiện tại. Nếu tìm thấy một đỉnh kề mà khoảng cách từ đỉnh nguồn đến đỉnh kề này thông qua đỉnh hiện tại nhỏ hơn khoảng cách đã biết, chương trình cập nhật khoảng cách mới và thêm đỉnh kề vào hàng đợi ưu tiên. Quá trình này tiếp tục cho đến khi không còn đỉnh nào trong hàng đợi ưu tiên hoặc đã duyệt qua tất cả các đỉnh trong đồ thị.

In ra khoảng cách từ đỉnh nguồn: Sau khi thuật toán hoàn thành, chương trình in ra khoảng cách từ đỉnh nguồn đến các đỉnh còn lại trong đồ thị.

Cách xử lý của chương trình này đảm bảo rằng khoảng cách từ đỉnh nguồn đến các đỉnh khác được tính toán một cách chính xác và tối ưu nhất theo thuật toán Dijkstra.

## Hướng dẫn thực thi chương trình

Để thực thi chương trình trong hàm main(), bạn có thể làm theo các bước sau:

1. Mở trình biên dịch C++ của bạn.

2. Tạo một tệp tin mới và sao chép đoạn code vào tệp tin đó.

3. Biên dịch mã nguồn thành tệp tin thực thi bằng lệnh: g++ dijkstra.cpp -o dijkstra.

4. Chạy chương trình bằng cách gọi tên tệp tin thực thi trong hàm main():

5. Biên dịch và chạy tệp tin thực thi bằng lệnh: ./dijkstra.

6. Chương trình sẽ chạy và thực hiện thuật toán Dijkstra, sau đó xử lý kết quả theo logic bạn đã viết trong hàm main().

.