

Tối ưu hệ thống đường đi giao hàng

Giới thiệu:

Công ty giaohangnhanh.vn có nhu cầu cải tiến tổng thể hệ thống giao hàng, trong đó việc tìm đường đi tối ưu được đặt lên hàng đầu. Bạn là chuyên gia công nghệ thông tin, có kiến thức chuyên sâu về thuật toán tìm đường, hãy thiết kế một chương trình mô phỏng việc tìm đường đi tối ưu để giới thiệu giải pháp công nghệ thông tin này cho giaohangnhanh.vn.

Mô tả bản đồ:

Mỗi địa điểm có số thứ tự và tên riêng. Các địa điểm được kết nối với nhau bởi đường đi có tên, chiều dài, và tốc độ trung bình xe có thể di chuyển trên đó.

Yêu cầu giải pháp:

Ở phiên bản đầu tiên (a), bạn hãy xây dựng một chương trình cho phép nhân viên giao hàng của giaohangnhanh.vn có thể tìm đường đi tối ưu từ địa điểm S(start) tới địa điểm E(end).

Input và output:

Dữ liệu bản đồ bạn được trung tâm dữ liệu bản đồ quốc gia cung cấp qua 2 tập tin văn bản có cấu trúc như sau:

`places.map` chứa tên các địa điểm, dòng thứ k chứa tên của địa điểm thứ k

`streets.map` chứa ma trận mô tả đường đi kết nối các địa điểm với nhau

Dòng đầu chứa n là số địa điểm

n dòng tiếp theo chứa ma trận mô tả đường đi

Mỗi dòng chứa n phần tử, mỗi phần tử cách nhau bởi ký hiệu '/', mỗi phần tử gồm có cấu trúc [Streetname,Length,Averagespeed] (Length: km, Averagespeed: km/h)

Ví dụ:

Nguyen Thi Minh Khai,8,3/Nguyen Dinh Chieu,7,2.5/Dien Bien Phu,15,3.5

Nếu không có đường đi: "-,-,-"

(a) Phiên bản 1.0 (1 điểm)

Viết chương trình theo tham số dòng lệnh với cấu trúc gọi chương trình như sau:

`[MSSV]_v1.exe [places.map] [streets.map] [start] [end] [prior] [path.txt]`

(b) Phiên bản 2.0 (2 điểm)

Giả sử, ứng dụng bạn đề xuất ở câu (a) đã được giaohangnhanh.vn yêu thích và yêu cầu bạn giới thiệu tính năng cao hơn: cho phép nhân viên tìm đường đi từ S tới E đi qua M(iddle). Bạn hãy viết chương trình theo tham số dòng lệnh với cấu trúc gọi chương trình như sau:

[MSSV]_v2.exe [places.map] [streets.map] [start] [middle] [end] [prior] [path.txt]

Giải thích tham số cho (a) và (b):

[MSSV]_[vx].exe: chương trình bắt buộc đặt tên theo MSSV

places.map: đường dẫn tập tin chứa danh sách địa điểm

streets.map: đường dẫn tập tin chứa sự liên kết giữa các địa điểm

start: địa điểm xuất phát

middle: địa điểm bắt buộc đi qua

end: địa điểm đến

prior: ưu tiên tìm đường nhanh hoặc ngắn nhất. 0: nhanh nhất, 1: ngắn nhất

path.txt: đường dẫn tập tin chứa đường đi từ đỉnh xuất phát tới đỉnh đến.

Dòng đầu chứa thời gian di chuyển nếu prior là 0 hoặc chiều dài đường đi nếu prior là 1. Các dòng tiếp theo là các đoạn đường đi, mỗi dòng là một đoạn có cấu trúc: [tên địa điểm] " --- " [tên đường] " ---> " [tên địa điểm].

(c) Phiên bản 3.0 (3 điểm)

Giaohangnhanh.vn muốn cung ứng dịch vụ chuyển hàng loạt thư đến một khu vực dân cư nằm trên nhiều con đường liên thông nhau. Bạn hãy xây dựng tính năng hỗ trợ nhân viên giao thư tính toán thứ tự đường đi để có thể đi qua hết tất cả con đường và mỗi con đường đi qua 1 lần. Tính năng này tự động tính toán địa điểm xuất phát và địa điểm kết thúc trong khu vực giao thư.

Tính năng này được thực hiện qua tham số dòng lệnh với cấu trúc như sau:

[MSSV]_v3.exe [places.map] [streets.map] [gps.map] [start] [streets_to_visit.map] [end] [route.map]

places.map, streets.map, start, và end được mô tả ở phần giải thích tham số cho (a) và (b).

gps.map: tập tin chứa tọa độ GPS của các địa điểm, dòng thứ k chứa tọa độ GPS của địa điểm thứ k. Mỗi dòng có cấu trúc: "Long,Lat". Ví dụ: "10.093674,108.932567".

streets_to_visit.map: tập tin chứa danh sách các con đường phải đi qua. Mỗi dòng là một con đường gồm [chỉ số của địa điểm đầu] [khoảng trắng] [chỉ số địa điểm cuối].

Lưu ý: người giao thư đi bộ nên có thể đi theo 2 chiều trên các con đường này, không quan tâm đường này 1 chiều hay 2 chiều.

route.map: tập tin chứa thứ tự đường đi đề xuất cho người giao thư: từ start đến địa điểm đầu của khu vực giao thư, đi qua hết các con đường trong khu vực giao thư, từ địa điểm cuối của khu vực giao thư tới end có cấu trúc như tập tin path.txt ở câu (a) và (b)

Địa điểm đầu của khu vực giao thư gần với start, địa điểm cuối của khu vực giao thư gần với end (dựa vào khoảng cách Euclide để xác định độ gần xa).

Ghi chú:

- Số địa điểm $n \leq 100$

- Start, end được đánh thứ tự từ 0 tới n-1
- Làm tròn thời gian hoặc chiều dài theo 5 chữ số thập phân.

(d) Tạo dữ liệu bản đồ (1 điểm)

Chuyển đổi dữ liệu bản đồ của maps.google.com của 1 quận Q (từ quận 1 – quận 10) trong Tp. HCM thành dữ liệu có cấu trúc trong đồ án này: [places_district_xx.map](#), [gpss_district_xx.map](#), [streets_district_xx.map](#), và [streets_to_visit_xx.map](#).

Q = chữ số cuối của MSSV, nếu chữ cuối của MSSV là 0 thì Q = 10.

xx là số thứ tự của quận: 01, 02, ..., 10

Lưu ý e1: không cần chuyển đổi toàn bộ dữ liệu, mỗi quận có từ 10 - 50 địa điểm và 20 – 100 con đường.

Lưu ý e2: streets_to_visit.map chứa các cạnh tạo thành đồ thị chứa đúng 2 đỉnh bậc lẻ

Lưu ý e3: các sinh viên có chữ số cuối của MSSV giống nhau có thể cùng nhau làm chung 1 bộ dữ liệu bản đồ. Số địa điểm = số sinh viên x 10, tối đa là 50. Số con đường = số sinh viên x 20, tối đa là 100. Trong trường hợp này, nhóm sinh viên cần mô tả rõ trong bài nộp.

(e) Tạo bộ test để kiểm tra chương trình (1 điểm)

Ứng với mỗi phiên bản v1, v2, và v3, tạo 5 bộ test để kiểm tra chương trình.

Mỗi bộ test gồm mã lệnh gọi chương trình và dữ liệu cần để chạy chương trình.

Mã lệnh được mô tả trong tập tin Cmds.txt: 5 dòng đầu là 5 bộ test của v1, 5 dòng kế là 5 bộ test của v2, và 5 dòng cuối là 5 bộ test của v3.

(f) Tính năng nâng cao (cộng 1 điểm cho điểm thực hành)

Thể hiện bản đồ và minh họa đường đi tìm được bằng giao diện đồ họa

Gợi ý: tìm hiểu MFC và GDI trong C++.

Nộp bài:

File nộp bài dạng [MSSV].zip. Nộp dung nộp gồm source code và file chương trình .exe

Cấu trúc thư mục nộp:

- [MSSV]_v1.exe
- [MSSV]_v2.exe
- [MSSV]_v3.exe
- [MSSV].sln
- [MSSV]_v1: chứa Project của câu a
- [MSSV]_v2: chứa Project của câu b
- [MSSV]_v3: chứa Project của câu c

- Map: thư mục chứa dữ liệu bản đồ ([places_district_xx.map](#), [streets_district_xx.map](#), [streets_to_visit_xx.map](#), và [contribution.txt](#)-chứa danh sách sinh viên tham gia tạo dữ liệu)
- Test: thư mục chứa dữ liệu test, gồm tập tin Cmds.txt và tập tin dữ liệu cần thiết để chạy lệnh test. Tập tin dữ liệu có thể chia sẻ giữa các mã lệnh chạy chương trình.
- Advance: thư mục chứa chương trình, mã nguồn, và dữ liệu bản đồ cho câu (f),

Xóa file trung gian trong 3 thư mục source, chỉ giữ lại file mã nguồn cần thiết giúp biên dịch được Solution. Giới hạn file nộp trên moodle: 1MB.

File trung gian: thư mục "Debug", "ipch", và file Netlist ".sdf"

Thời gian làm bài: 2 tuần

Deadline: 23:55 17/12/2015

Lưu ý:

Chép bài hoặc để bạn chép bài sẽ được 0 điểm phần thực hành môn học.

Đồ án cuối kỳ được mở rộng từ đồ án giữa kỳ (câu a và b). Do đó 2 đồ án sẽ được gộp lại chấm điểm chung. Sinh viên nào không nộp đồ án giữa kỳ sẽ bị trừ 1 điểm.

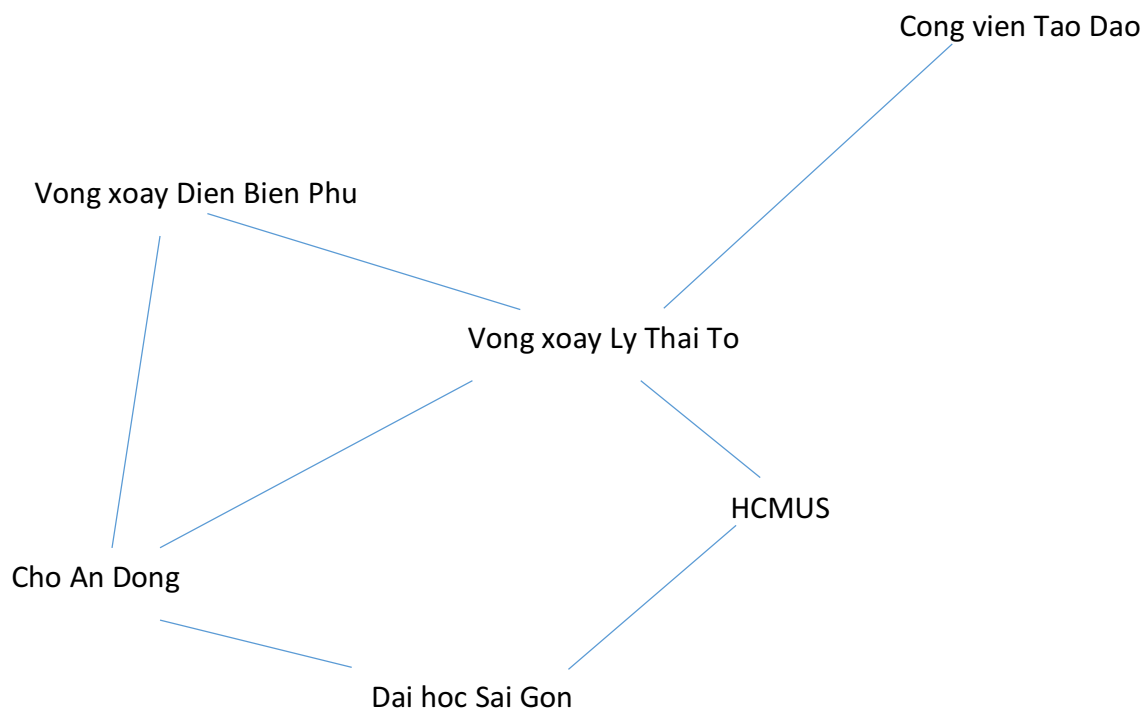
Phụ lục:

Mẫu tập tin chứa địa điểm places.txt

HCMUS
Vong xoay Ly Thai To
Vong xoay Dien Bien Phu
Cong vien Tao Dan
Cho An Dong
Dai hoc Sai Gon

Mẫu tập tin chứa đường liên kết các địa điểm streets.txt

6
-,-,-/Nguyen Van Cu,1,2/-,-,-/-,-,-/-,-,-/Nguyen Van Cu,2,2
Nguyen Van Cu,1,2/-,-,-/Ly Thai To,3,6/Nguyen Thi Minh Khai,5,4/Hung Vuong,4,3/-,-,-
-,-,-/Ly Thai To,3,3/-,-,-/-,-,-/-,-,-/Ngo Gia Tu,2,5,10/-,-,-
-,-,-/Nguyen Thi Minh Khai,5,4/-,-,-/-,-,-/-,-,-/-,-,-
-,-,-/-,-,-/-,-,-/Ngo Gia Tu,2,5,8/-,-,-/-,-,-/-,-,-/Nguyen Trai,3,1
Nguyen Van Cu,2,2/-,-,-/-,-,-/-,-,-/Nguyen Trai,3,5/-,-,-



Hình 1. Minh họa bản đồ khu vực xung quanh HCMUS