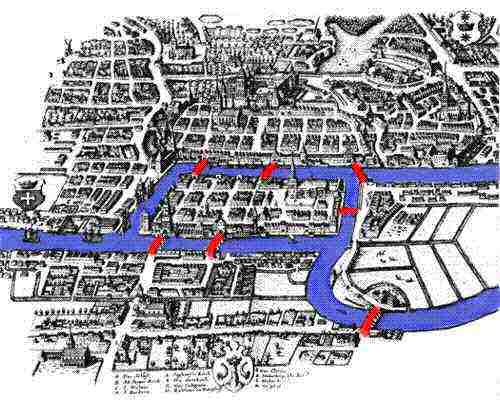
**phần Giải thuật Euler để ở trên định nghĩa 1 khúc**

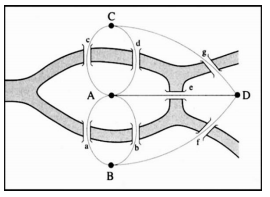
Bài toán

Có thể xuất phát tại một điểm nào đó trong thành phố, đi qua tất cả 7 cây cầu, mỗi cây một lần, rồi trở về điểm xuất phát được không?



Leonhard Euler đã tìm ra lời giải cho bài toán vào năm 1736

Từ hình ở thành phố, ta có thể đơn giản hóa lại bài toán này thành 1 đồ thị graph như sau:



Hình chia thành 4 bản đất chính đó là A, B, C, D khác nhau, ta coi mỗi chính là 1 đỉnh tương ứng, còn 7 câu cầu còn lại nối các mảnh đất đó, ta gọi chúng là cạnh.

Đáp án chính xác mà Euler đề ra đó là KHÔNG. Ta không thể đi qua tất cả các con cầu trên và trở lại vì trí ban đầu, mà mỗi cầu chỉ đi qua được 1 lần.

Tại sao ông Euler lại nói như vậy, ừ thì ta có thể thử nghiệm đi từ từng đỉnh qua mỗi cạnh, rùi nếu không thành thì ta nói là mình không đi qua được, nhưng điều này chỉ đúng với những bài có ít cạnh và đỉnh, do đó mà ông Euler đã cho ra nhưng định lí mà ta có thể nhìn ra được ngay, trong 1 đồ thì nào đó, có thể mình dùng được đường đi Euler để đi qua các con đường được hay không.

Điều kiện cần để tồn tại chu trình Euler là:

* + Trong đồ thị vô hướng, bậc của tất cả các đỉnh phải là số chẵn.
  + Trong đồ thị có hướng, bậc ngoài và bậc trong của mỗi đỉnh phải bằng nhau

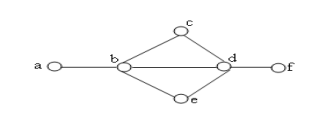
Trong một đồ thị, nếu không tìm ra chu trình Euler, vẫn có thể tồn tại đường đi Euler. Điều kiện cần để tồn tại đường đi Euler trong trường hợp này là:

* + Trong đồ thị vô hướng, tồn tại duy nhất hai đỉnh có bậc lẻ, tất cả các đỉnh còn lại là bậc chẵn.
  + Trong đồ thị có hướng, bậc ngoài và bậc trong của mỗi đỉnh bằng nhau ngoại trừ một đỉnh tại đó bậc ngoài lớn hơn bậc trong 1 đơn vị (làm đỉnh bắt đầu) và một đỉnh tại đó bậc trong lớn hơn bậc ngoài 1 đơn vị (làm đỉnh kết thúc).

**Khúc này viết dưới phần định nghĩa.**

Còn 1 bài toán thực tiễn cũng được áp dụng nhiều ở thực tiễn, đó là Chinese Postman:

Cho 6 ngôi nhà cách nhau 1 khúc đường mà người đưa thư cần phải đến, bản đồ các ngôi nhà nối vối nhau bằng con đường nhựa được sắp như sau:



Từ đồ thì mà người dao hang vẽ ra, ta đặt cho mỗi nhà là 1 đỉnh của 1 đồ thị, gọi từng ngôi nhà là 1 đỉnh tương ứng, A, B, C, D, E, F. Người dao hàng bắt đầu từ từ góc ngồi nhà xa nhất, đó là ngôi nhà ở vì trí A, đi giao hang tới góc nhà xa nhất phía bên kia, đó là ngôi nhà ở vị trí F. Nhưng nếu mà mà người gia hang này không tính toán trước, mà đi một mạch tới F, thì người giao hàng sẽ phải quay lại mà giao hàng cho các ngôi nhà khác, làm phải đi lại quảng đường ít nhất là đi lại quãng đường (D,F). Vậy nên nếu mà người giao hàng áp dụng giải thuật Euler, thì người giao hàng này có thể đi qua tất cả các ngôi nhà mà không phải tốn thêm thời gian nào đi lại quãng đường, và đó chính là quãng đường đi từ A, B, C, D, B, E, D, F.

**Ứng dụng Hamilton để ở sau Định nghĩa**

Giãi thuật Hamilton cũng có rất nhiều ứng đụng thực tiến trong đời sống chúng ta, Ví dụ như trong 1 vi mạch điện, Hamilton giúp ta tìm được đường đi cho mạch điện mình không đi đụng chạp tới các đỉnh đã được nối rồi.

Ví dụ 2: Một ứng dụng rất đơn giản là lập kế hoạch tuyến xe buýt để đón học sinh (nút-> học sinh, đường-> cạnh, đường xe buýt-> đường Hamilton) từ đó mà chủ xe bus không có lập lạị để đi tới 1 khu nào đó không có học sinh đứng đón xe bus.

Nó được sử dụng trong lập bản đồ bộ gen để kết hợp nhiều đoạn mã di truyền nhỏ.

**Bài toán sử dụng prim’s algorithms và krushul viết sau trên giới thiệu 2 thuật toán định nghĩa**

Ví dụ như 1 khu làng có nhiều nhà nào đó muốn có mạng truyền hình để xem TV, và ta muốn

chi phí thiết lập toàn bộ kết nối dây cáp đến cả ngôi làng đó để được tối thiểu giảm chi phí. Vậy nên chúng ta phải nên dùng thật toán Prim hoặc là thuật toán Kruskal

Cả hai thuật toán cũng chỉ là hai nhánh tương tự của một cây bao trùm tối thiểu. Cả hai đều có logic dễ dàng, các trường hợp xấu nhất giống nhau và sự khác biệt duy nhất là việc triển khai có thể liên quan đến cấu trúc dữ liệu khác một chút. Vậy đâu là yếu tố quyết định? Cái khác biệt cơ bản nhất là cách bạn chọn đỉnh hoặc là cạnh của đồ thị ban đầu khi ta muốn tìm cây khung nhỏ nhất

**Cây khung nhỏ nhất**

Cây khung nhỏ nhất trong một đồ thị liên thông, có trọng số là một cây khung có tổng trọng số trên các cạnh của nó là nhỏ nhất.

Cây khung nhỏ nhất

Thuật toán Kruskal

Bắt đầu bằng việc chọn một cạnh có trọng số nhỏ nhất, đặt nó vào cây khung T.

Trong khi cây khung T có ít hơn (n-1) cạnh

Ghép vào T cạnh có trọng số nhỏ nhất và không tạo ra chu trình trong T.

Chìa khóa để lựa chọn giữa thuật toán của Prim’s và Kruskal cho một ứng dụng là nghiên cứu cấu trúc của chính ứng dụng đó.

Thuật toán của Prim nhanh hơn đáng kể trong giới hạn khi bạn có một đồ thị thực sự dày đặc (nhiều cạnh hơn nhiều đỉnh). Kruskal hoạt động tốt hơn trong các tình huống điển hình (đồ thị thưa thớt) và dễ thực hiện hơn vì nó sử dụng các bộ rời rạc và cấu trúc dữ liệu đơn giản hơn.

Thường được sử dụng ở đây là nên làm, chứ vẫn có thể dùng thuật toán Prim áp dụng cho những ứng dụng mà Kruskal thường dùng và ngược lại được.

Cho nên, các ứng dụng mà thuật toán của kruskal thường được sử dụng:

1. Cáp hạ cánh

2. Mạng TV

3. Điều hành Tour

4. Mạng LAN

5. Mạng lưới đường ống dẫn nước uống hoặc khí đốt tự nhiên.

6. Một lưới điện

7. Cụm liên kết đơn

Các ứng dụng mà thuật toán của prim thường được sử dụng:

1. Mạng lưới đường bộ và đường sắt kết nối tất cả các thành phố.

2. Kênh tưới và đặt tháp vi ba

3. Thiết kế lưới cáp quang hoặc IC.

4. Bài toán Nhân viên Bán hàng Đi du lịch.

5. Phân tích cụm.

6. Các thuật toán tìm đường được sử dụng trong AI (Trí tuệ nhân tạo).

7. Phát triển trò chơi