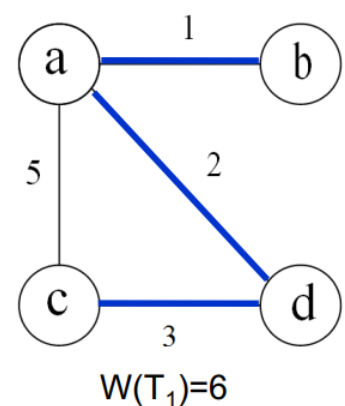
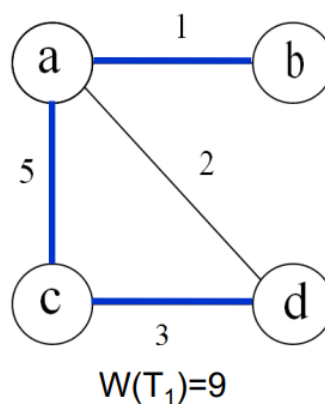
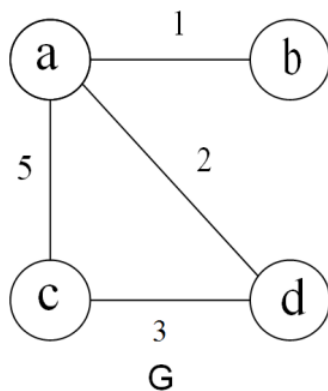


Cây bao trùm nhỏ nhất

I. Bài toán cây bao trùm tối thiểu

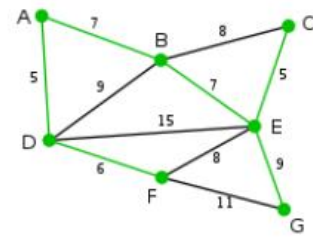
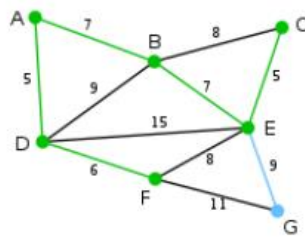
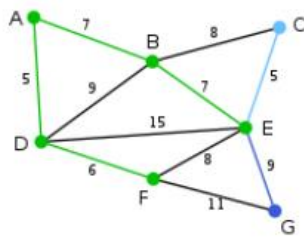
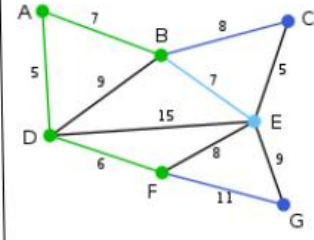
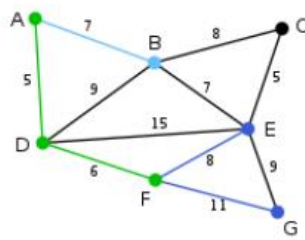
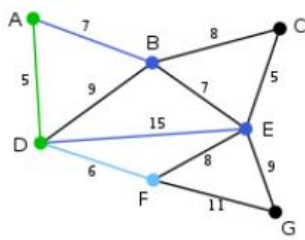
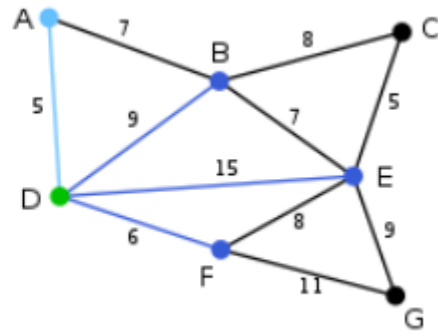
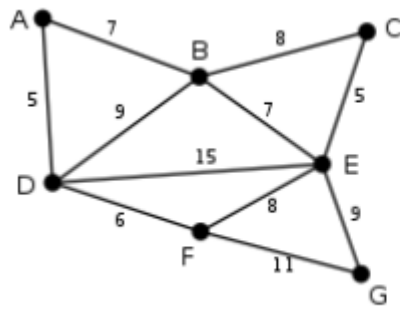
Cây bao trùm (Spanning Tree) của đồ thị G là một đồ thị con liên thông, không có chu trình (cây) có chứa tất cả các đỉnh của G .

Cây bao trùm tối thiểu (Minimum Spanning Tree) của một đồ thị liên thông có trọng số G là cây bao trùm có tổng trọng số các cạnh là nhỏ nhất.



II. Thuật toán Prim

- Bắt đầu với cây chỉ có 1 đỉnh T_0 .
- Phát triển cây theo các bước, mỗi bước thêm một đỉnh vào cây đã có bằng một cạnh. Dãy các cây được phát triển T_1, T_2, \dots, T_{n-1} .
- Chiến lược tham lam: Tại mỗi bước dựng cây T_{i+1} từ cây T_i với việc thêm đỉnh “gần nhất”.
- Đỉnh gần nhất với T_i : Đỉnh không thuộc T_i và được nối với T_i bằng cạnh có trọng số nhỏ nhất.
- Thuật toán dừng lại khi tất cả các đỉnh đã được thêm vào.



- **Lược đồ thuật toán Prim**

$\text{Prim}(G) \equiv$

// Input: $G = (V, E)$

// Output: E_T , tập các cạnh của cây bao trùm tối thiểu của G

$V_T = \{v_0\}$

$E_T = \emptyset$

for $i=1..|V| - 1$

 tìm cạnh có trọng số nhỏ nhất $e^*=(v^*,u^*)$ trong tất cả

 các cạnh (v, u) mà $v \in V_T$ và $u \in V - V_T$;

$V_T = V_T \cup \{u^*\}$

$E_T = E_T \cup \{e^*\}$

endf;

return E_T ;

End.

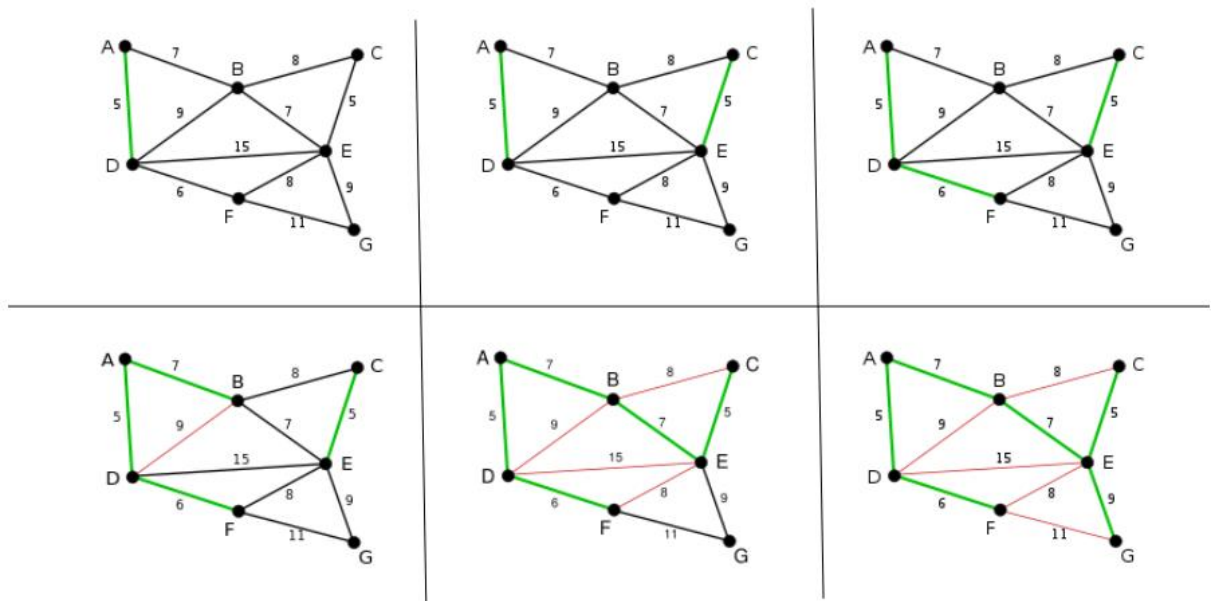
- Chứng minh tính đúng đắn

Sử dụng phương pháp quy nạp và phản chứng

- Chứng minh bằng quy nạp: mỗi cây con T_i , $i = 0, \dots, n-1$, sinh ra bằng thuật toán Prim là một phần của cây bao trùm tối thiểu. Giả sử T_{i-1} là một phần của cây bao trùm tối thiểu thì T_i (sinh ra bằng cách thêm vào cạnh gần nhất) cũng là một phần của cây bao trùm tối thiểu.
- Phản chứng: giả sử cây bao trùm tối thiểu của G không chứa T_i và có cạnh $e_i = (v, u)$ là cạnh có trọng số nhỏ nhất nối một đỉnh thuộc T_{i-1} với 1 đỉnh không thuộc T_{i-1} , và e_i không thuộc cây tối thiểu T (theo Prim).
- Nếu thêm e_i vào T sẽ tạo ra một chu trình, chu trình sẽ chứa cạnh (v', u') nối một đỉnh $v' \in T_{i-1}$ với 1 đỉnh $u' \notin T_{i-1}$. Nếu xóa đi cạnh (v', u') của chu trình này sẽ nhận được một cây bao trùm khác chứa e_i có trọng số nhỏ hơn T . Như vậy trái với giả thiết cây bao trùm tối thiểu không chứa $T_i \Rightarrow$ đpcm

III. Thuật toán Kruskal

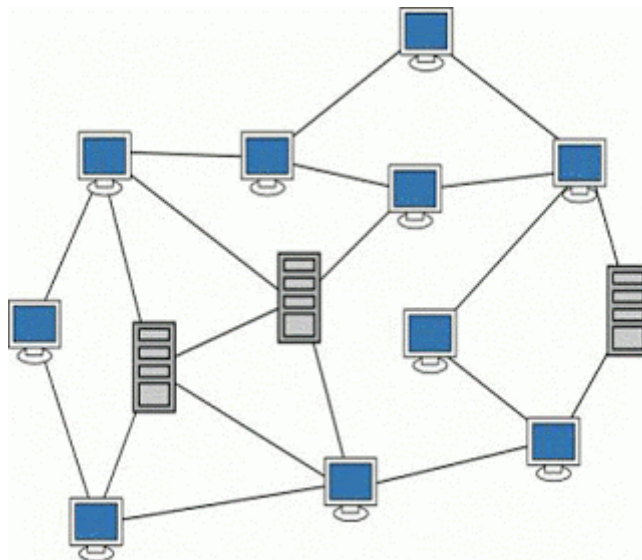
- Các cạnh được sắp xếp theo thứ tự tăng dần của trọng số.
- Bắt đầu bằng 1 rừng cây (forest) rỗng.
- Xây dựng MST theo các bước, mỗi bước thêm một cạnh.
 - Trong quá trình dựng MST luôn có một “rừng”: các cây không liên thông.
 - Thêm vào cạnh có trọng số nhỏ nhất trong các cạnh chưa thêm vào cây và không tạo thành chu trình.
 - Như vậy tại mỗi bước một cạnh có thể:
 - Mở rộng một cây đã có.
 - Nối hai cây thành 1 cây mới
 - Tạo cây mới
- Thuật toán dừng lại khi tất cả các đỉnh đã được thêm vào.



- Chứng minh tính đúng đắn của thuật toán:
-

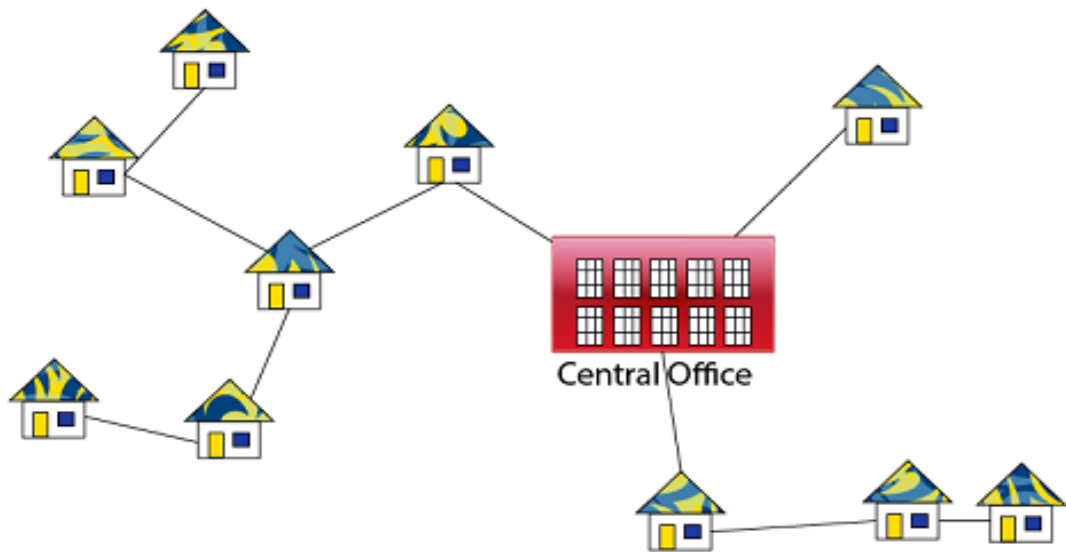
IV. Ứng dụng của cây bao trùm nhỏ nhất

- **Thiết kế mạng:** Cây bao trùm tối thiểu có các ứng dụng trực tiếp trong thiết kế mạng, bao gồm cả mạng máy tính, mạng viễn thông, mạng giao thông, mạng lưới cấp nước và lưới điện.



Thiết kế đường mạng

Wiring : Better Approach



Minimize the total length of wire connecting the customers

- **Các thuật toán gần đúng cho các vấn đề NP-hard:** - vấn đề nhân viên bán hàng du lịch, cây Steiner Một ứng dụng ít rõ ràng hơn là cây bao trùm tối thiểu có thể được sử dụng để giải quyết xấp xỉ vấn đề nhân viên bán hàng du lịch. Một cách chính thức thuận tiện để xác định vấn đề này là tìm con đường ngắn nhất ghé thăm mỗi điểm ít nhất một lần.
- **Các ứng dụng gián tiếp:** - đường dẫn tắc nghẽn tối đa - Mã LDPC để sửa lỗi - đăng ký hình ảnh với entropy Renyi - học các tính năng nổi bật để xác minh khuôn mặt thời gian thực - giảm lưu trữ dữ liệu trong chuỗi axit amin trong protein - địa phương mô hình của các tương tác hạt trong dòng chảy hỗn loạn. - giao thức autoconfig cho kết nối Ethernet để tránh các chu kỳ trong mạng.
- **Phân tích cụm:** vấn đề phân cụm k có thể được xem là tìm MST và xóa các cạnh đắt nhất k-1

目前的概率密度估计方法主要分为参数估计和非参数估计
最大似然估计 无参估计 几个正态分布 混合而成的概率密度函数
长期以来人们仍寻求一种方法 只使用训练样本中的某些统计估计
在实际应用中 训练样本的数目往往是随人的选择影响
较大的 些样本 以减少计算量 缩短运算时间 增强实用性

~~In consideration of your use of the service, you represent that you are of legal age to form a binding contract and are not a person barred from receiving services under the laws of the United States or other applicable jurisdiction. You also agree to: (a) provide true, accurate, current and complete information about yourself as prompted by the service's registration form and (b) maintain and promptly update the registration data to keep it true, accurate, current and complete.~~

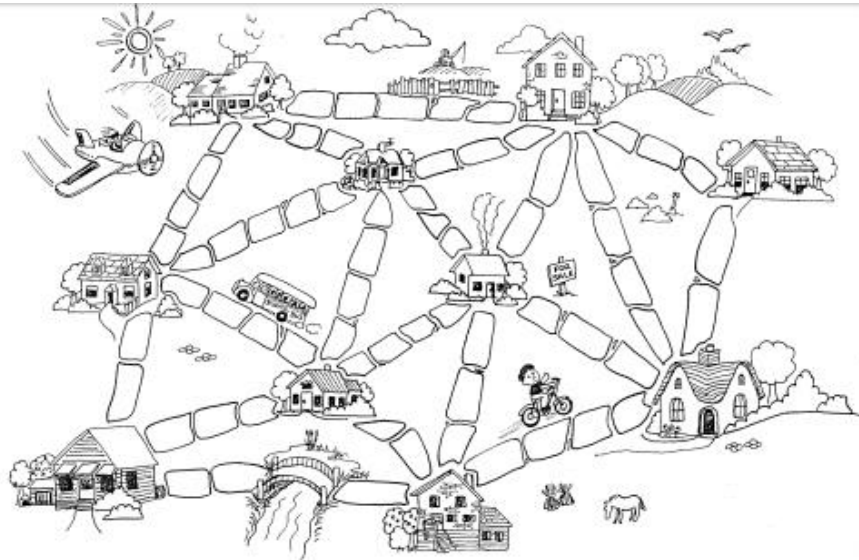
~~For this reason, parents of children under the age of 13 who wish to allow their children to access the service must create a family account.~~

- Bài toán cụ thể:

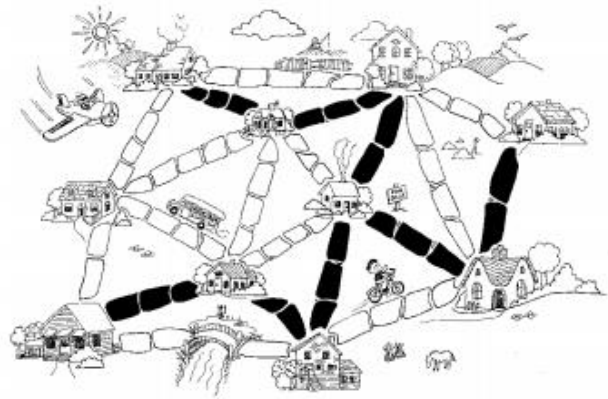
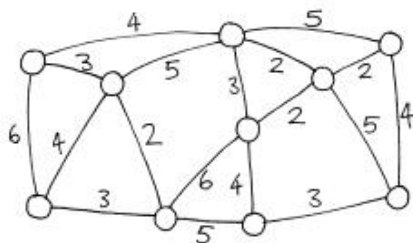
1. Việc lát nền nên chi phí càng ít càng tốt.

Dưới đây là cách bố trí của thành phố. Số lượng đá lát giữa mỗi ngôi nhà đại diện chi phí lát đường đó. Tìm tuyến đường tốt nhất kết nối tất cả các ngôi nhà, nhưng sử dụng ít quây (lát đá) càng tốt.

Giải pháp: biểu đồ (cho một thành phố bùn khác) và lát nền.



Solution: the graph (for another muddy city) and the paving.



2. <https://www.hackerearth.com/blog/developers/kruskals-minimum-spanning-tree-algorithm-example/>

