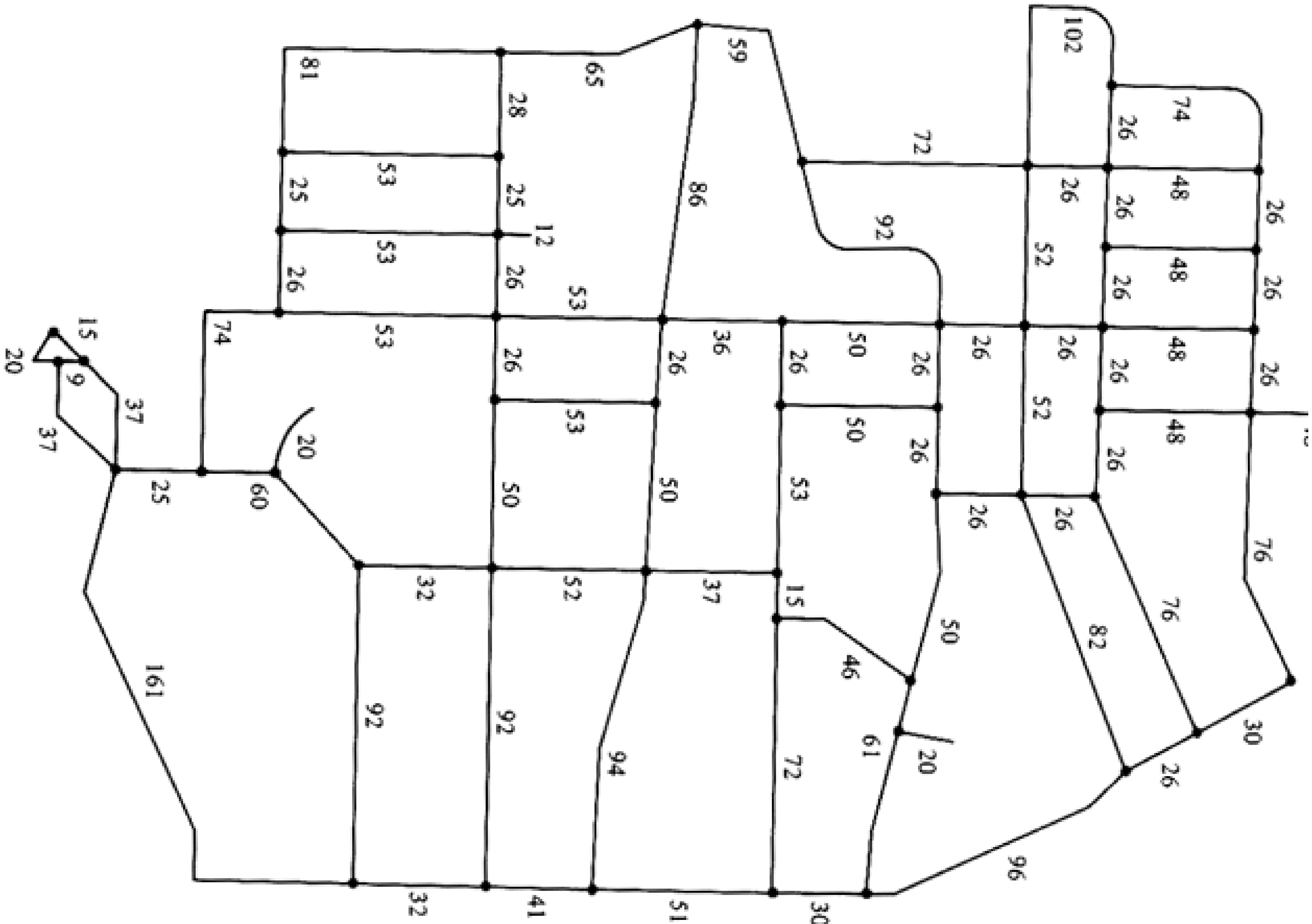
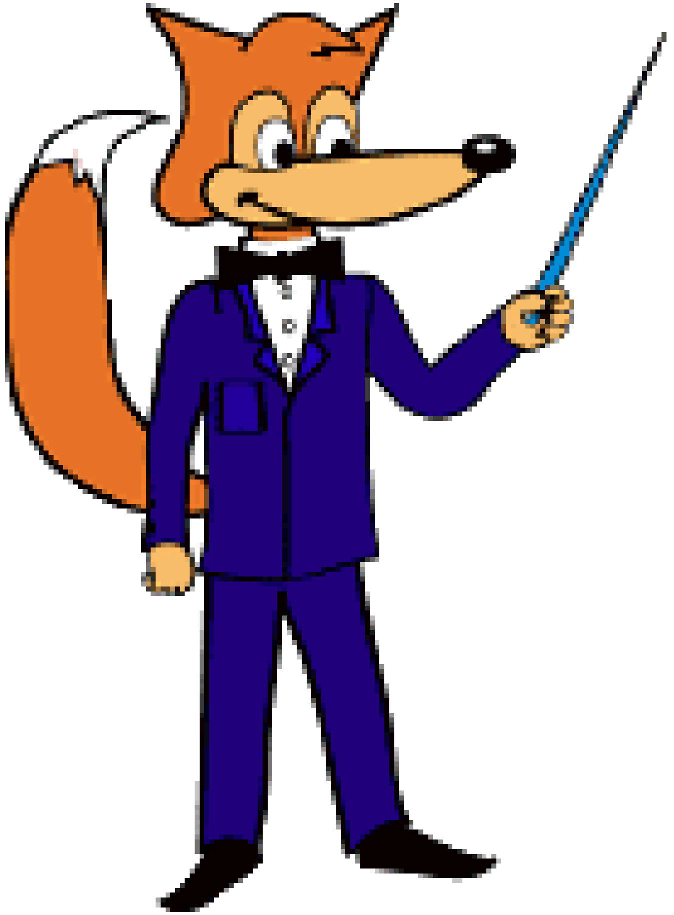
Graph Theory games and real life applications.





CONTENTS

- Graph Theory Definition
- The three types of Graphs
- Degree of the vertex
- Handshake Theorem
- Eulerian Graphs
- Weighted Graphs
- Algorithm of Chinese post man.
- Planar graphs and Euler formula

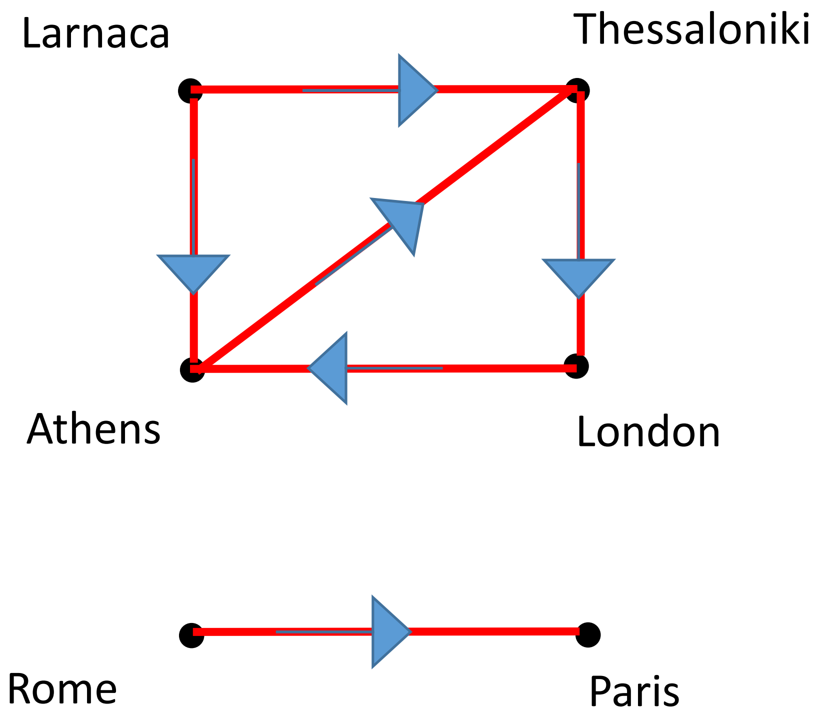
Graph theory:

Is the formal study of a symbolic representation, based on the connectivity of vertices and edges. (concerns the relationship among lines and points)

Many problems of practical interest can be represented by graphs.

Example: The following table shows the flights from various European airports.

Larnaca	Thessaloniki, Athens
Thessaloniki	Larnaca, London
London	Thessaloniki, Athens
Athens	Larnaca, Thessaloniki, London
Rome	Paris



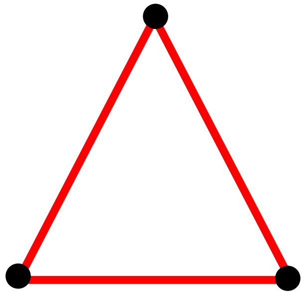
a) Is there a way to travel from Larnaca to Paris?

b) What do the vertices represent?

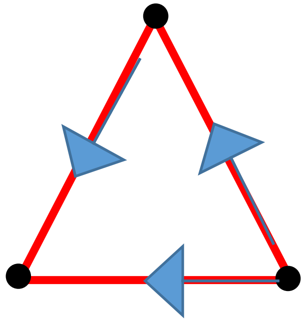
c) What do the edges represent?

There are three types of graphs:

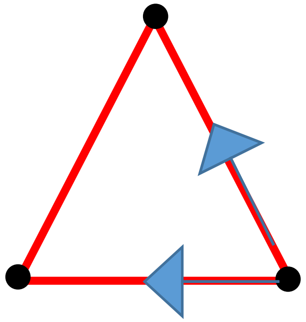
Undirected



Directed



Mixed graphs

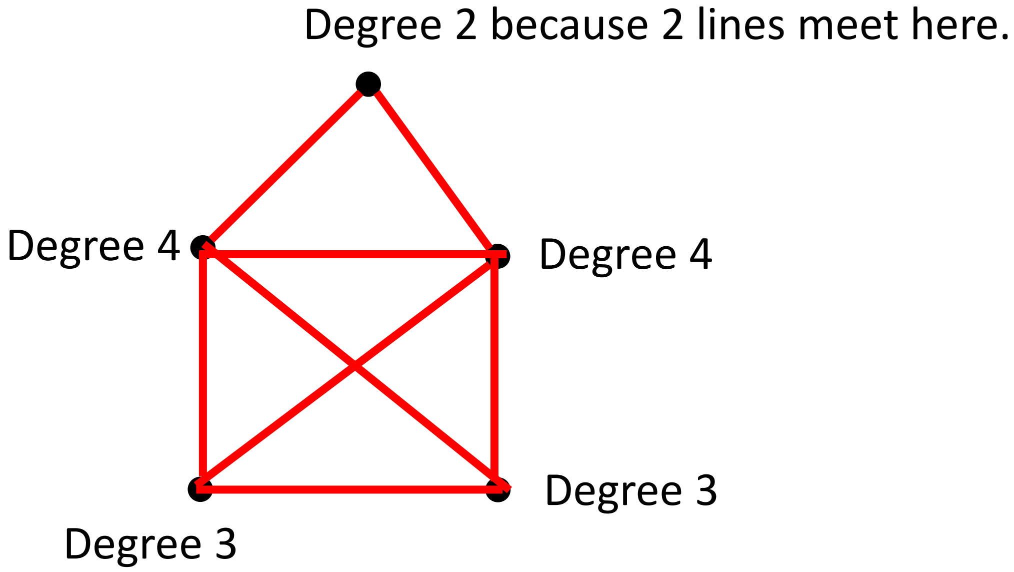


For example, if vertices represents people.

- Undirected: if the shake hands, if a person A shake hands with a person B, then person B also shook hand with person A.

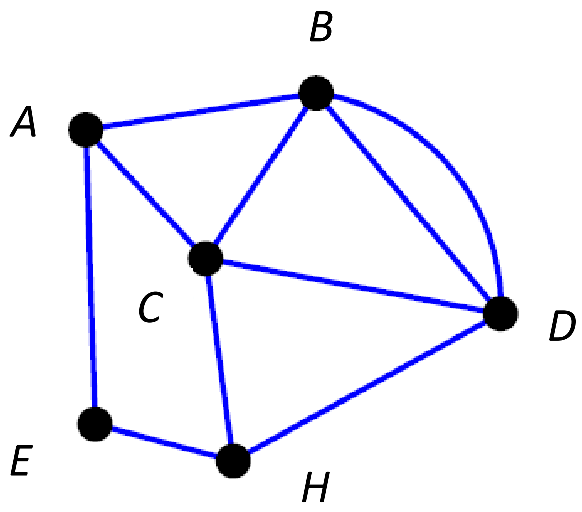
- Directed: if one person knows another, does not necessarily implies the reverse (one person is famous).

Degree of a vertex is the number of edges incident to it.



Exercise 1

a) Write down the degree of each vertex



Vertex	Degree
A	3
B	4
C	4
D	4
E	2
F	3

b) Write down the sum of the degrees of all vertices.

3+4+4+4+2+4=20

Exercise 2

a) Can you draw a graph with the sum of degree of vertices to be an odd number?

b) Is it possible for 5 people to shake hand with exactly 3 of them ?

Handshaking Theorem

Since every edge connect two vertices one for each of its end points, the total sum of degree of vertices must be twice the sum of the edges.

$$\text{Sum of edges} = \frac{\text{sum of degree of vertices}}{2}$$

Conclusions:

- Sum of degree of vertices must be an even number.
- Every graph has an even number of odd degree vertices.

Exercise 3

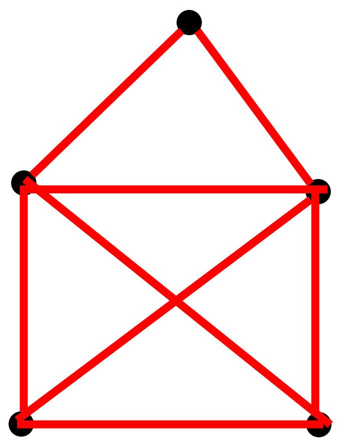
a) In an octagon, what is the sum of its edges, if we draw all of its diagonal?

b) Write down the general formula for the number of sides of a closed polygon if we join all its vertices.

Games for connected graphs

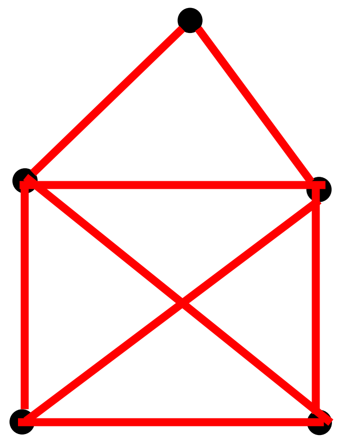
A graph is **connected** if there is a path for every pair of vertices.

Is it possible to go through each edge exactly once?



Eulerian path is a path on the graph that visits every edge exactly once. (start and ends at different vertices).

Is it possible to go through each edge exactly once, start and end on the same vertex?



Eulerian circuit is a Eulerian path which starts and ends on the same vertex.

Eulerian Path

Eulerian path is a path on the graph that visits every edge exactly once.
(start and ends at different vertices).

The key insight is that except for the starting and ending vertices, the path you are following must go through a vertex along one edge and then out of the same vertex along a different edge. Such a process always uses up two of the edges which derive from the vertex. We may go through the same vertex again any number of times, but each time we are eliminating exactly two edges. Thus each vertex which is not the starting or ending point has an even number of edges coming out of it.

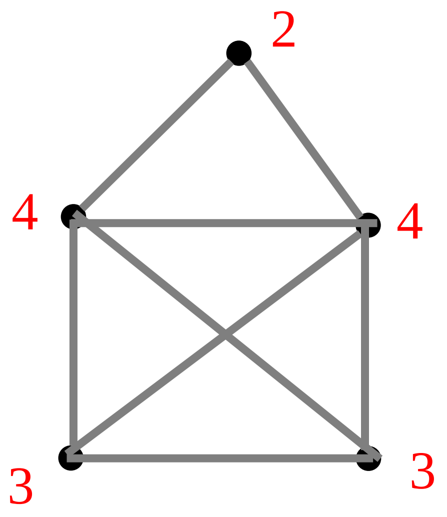
The starting and ending vertices, will be the only two vertices which must have an odd degree.

Therefore, a Eulerian path always exist if there are **exactly** two odd vertices, and an Eulerian path will always start and and end on the odd degree vertices.

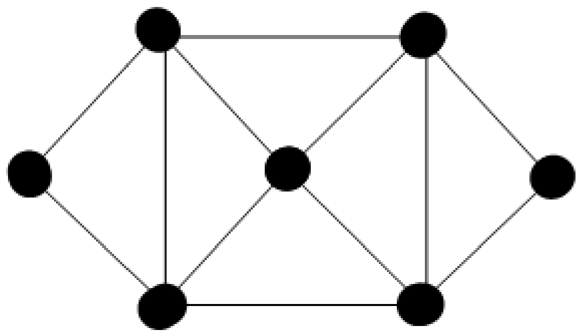
Exercise 4

a) Which of the following graphs have a Eulerian path?

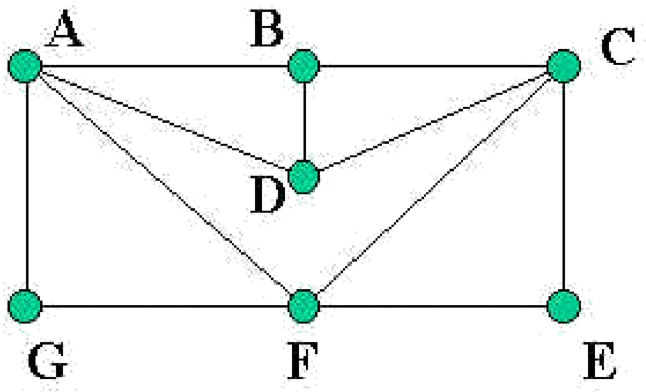
i)



ii)



iii)



b) For those graph that there exist a Eulerian path find it.

Eulerian Circuit

Eulerian circuit is a Eulerian path which starts and ends on the same vertex.

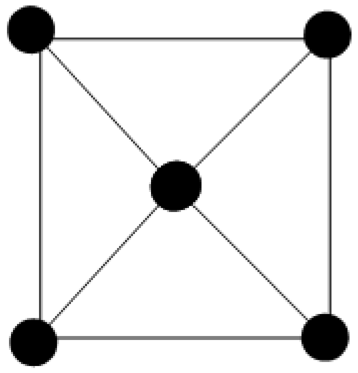
Based on observation we found from the Eulerian Path, if we want to end back on the same vertex, we need to add an edge between the two odd vertices. But since adding this edge increases the degree of each by 1, their degrees are then even.

Therefore, a Eulerian circuit always exist if all of its vertices are of even degree. In other words, a Eulerian circuit always exist if there are not odd degree vertices.

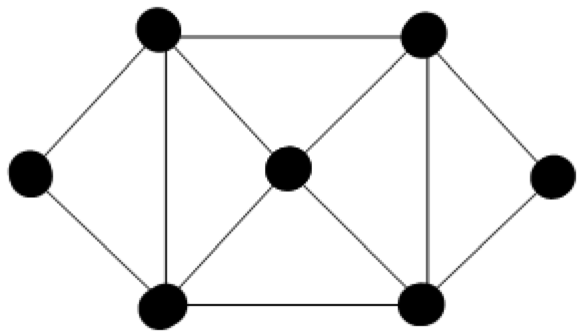
Exercise 5

Which of the following graphs have a Eulerian circuit?

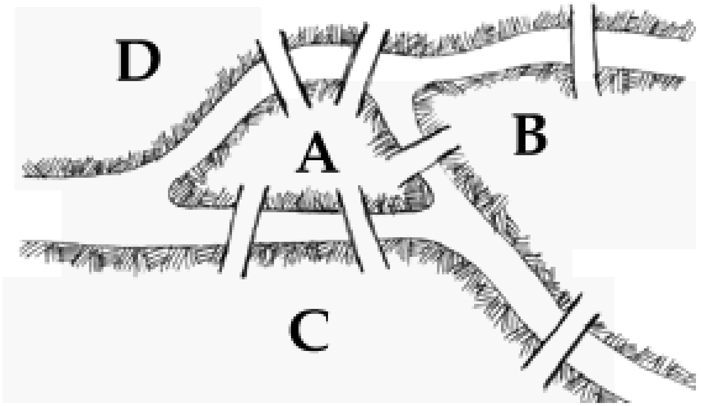
i)



ii)



iii)

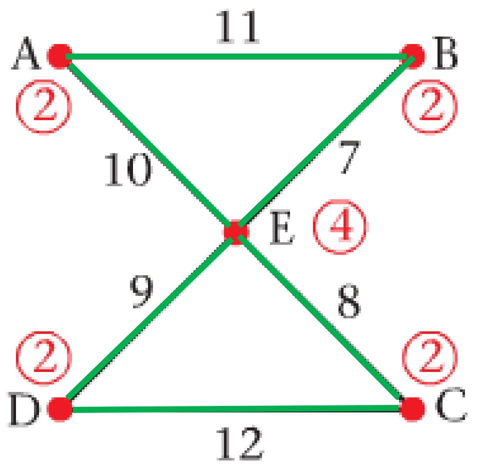


Next question: If an Euler path or circuit exists, how do you find it?

Try to always have a connected graph. In other words do not burn your **bridge**.

THE MINIMAL ROUTE FOR WEIGHTED NETWORKS

EULERIAN GRAPH



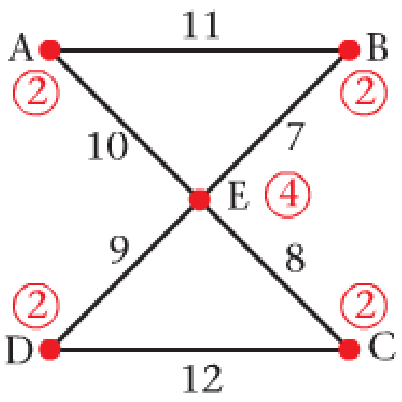
Graph 1

Distance between two vertices: The total length of an edge (Distance of edge AB = 11 and AB=BA)

Task 1: Find a minimal route for Graph 1, starting and finishing at A.

A possible route: ABECDEA

Other possible routes: ABEDCEA, AEDCEBA



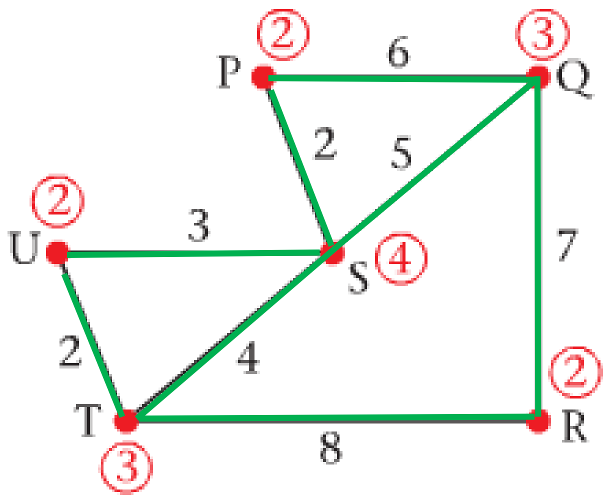
Graph 1

The total distance in Graph 1 is 11+7+8+12+9+10 = 57. Notice that the shortest route will be equal to the weight of the network

Notes: 1) It is possible to choose any vertex to be the starting or the ending point of the route.

2) For Weighted Eulerian graphs the weight of the network is always equal to the minimal route since the total distance is the same independently of the order we move from one vertex to another.

SEMI-EULERIAN GRAPHS



Graph 2

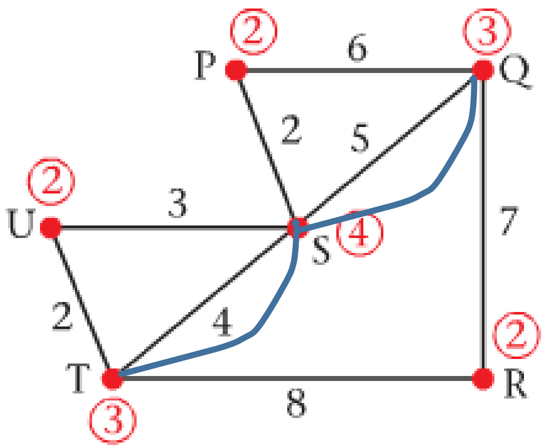
Notice that: 1) Vertices T and Q have odd valency (3) and all the remaining vertices have even degree.

Task 2: Find the minimal route for Graph 2 that traverses each edge only once.
It is permitted to start and finish at different vertices

One solution : TRQSTUSPQ **Minimal route: 37**

Comments: 1) We can find a solution to the above problem only if the starting and the ending points are the two vertices with odd degree.

2) The only way to find a route in this network by starting from a vertex of an even degree is by traverse an edge (or edges) at least twice. Therefore the total weight of the route is increased.



Graph 2

Task 3: Find a minimal route, starting and finishing at S, that traverses each edge at least once.
State your route and its length.

How to solve: The odd degrees are at vertices Q and T. We have to repeat the shortest path Q to T, which is Q S T, of length 9.
(Transforming the graph from semi – Eulerian to Eulerian)

Possible solution: SQPSQRTSUTS

Comments: 1) Using the same vertex to be the starting and the finishing point result the total route to be increased by the weight of the repeated path.

Total weight of minimal route: 37+9 =46

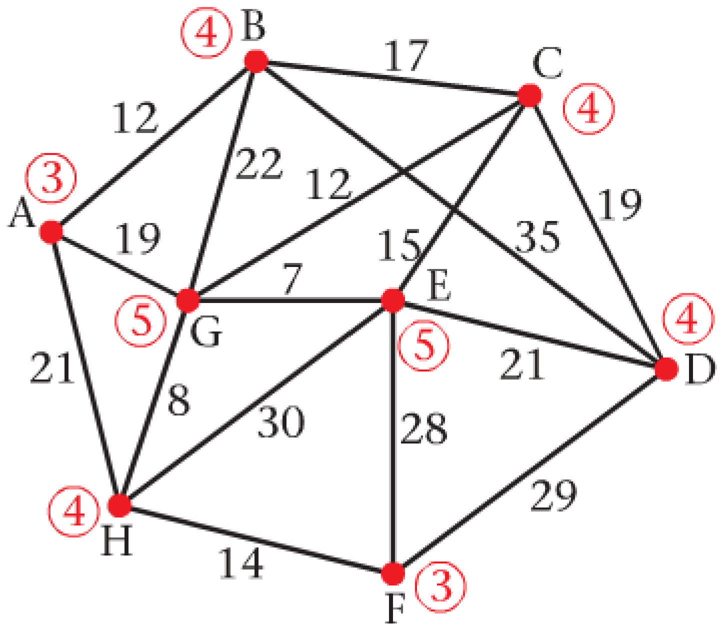
ROUTE INSPECTION ALGORITHM (CHINESE POSTMAN)

This algorithm can be used to find the shortest route that traverses every edge at least once and returns to the starting point.

HERE IS THE ROUTE INSPECTION ALGORITHM:

- Identify any vertices with odd degree.
- Consider all possible complete pairings of these vertices.
- Select the complete pairing that has the least sum.
- Add a repeat of the edges indicated by this pairing to the network.

APPLYING THE ROUTE INSPECTION ALGORITHM IN NON EULERIAN GRAPHS



Graph 3

Question 6: Solve the route inspection problem for this network starting and finishing at A.

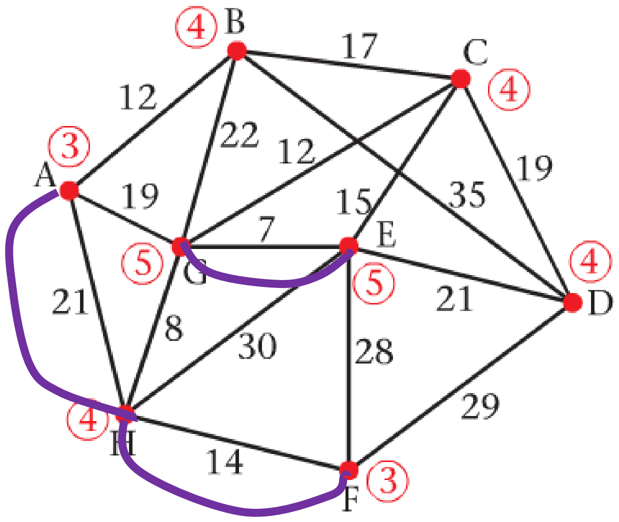
How to solve: 1) Identify any vertices with odd degree (A,E,F,G)

2) The shortest paths for all pairings of the four vertices:

AE+FG = 26+22 =48

AF+EG = 35+7 = 42 → Least sum

AG+EF = 19+28 = 47



Graph 3

3) We need to repeat edges AF (AH + HF) and EG to the network (Graph 3)

Possible solution: ABCDBGCEGAHEDFEGHFHA

Using A as the starting and finishing point the length of the shortest route will be:

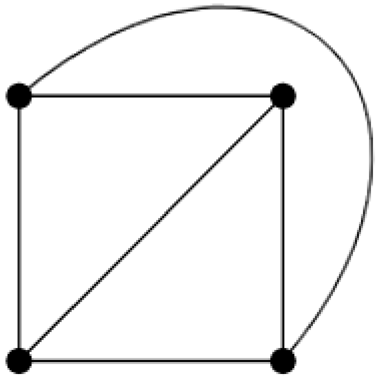
309 (weight of initial graph 3) + 42 (least sum of repeated edges) = 351

Planar graphs

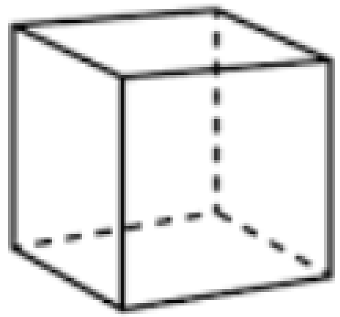
When is it possible to draw a graph so that none of the edges cross? If this *is* possible, we say the graph is ***planar*** (since you can draw it on the *plane*).

When a planar graph is drawn without edges crossing, the edges and vertices of the graph divide the plane into regions. We will call each region a ***face***. The number of faces does not change no matter how you draw the graph (as long as you do so without the edges crossing), so it makes sense to ascribe the number of faces as a property of the planar graph.

We count as a face the outside of the shape in 2D shape only.



Has 4 faces,
Count the outside



Has 6 faces,
Do not Count the outside in 3D

Example of a problem:

A railway station have suggested that some of the rail lines, stations and/or trains should be eliminated to allow the company to minimize its losses. A team of consultants has been hired to analyze the rail services that are available in any region and decide where it needs to concentrate its money and services to best meet the needs of its customers. As a team of consultants you will need to make recommendations in your area of expertise as to what the company should do.

The first step is encouraging the students to visualize the data by drawing train routes. Discuss with them what it means for cities to be connected. The mapped train routes can be transformed into formal graphs where vertices are cities and edges are route segments that connect the cities

Thinks which are important for the company:

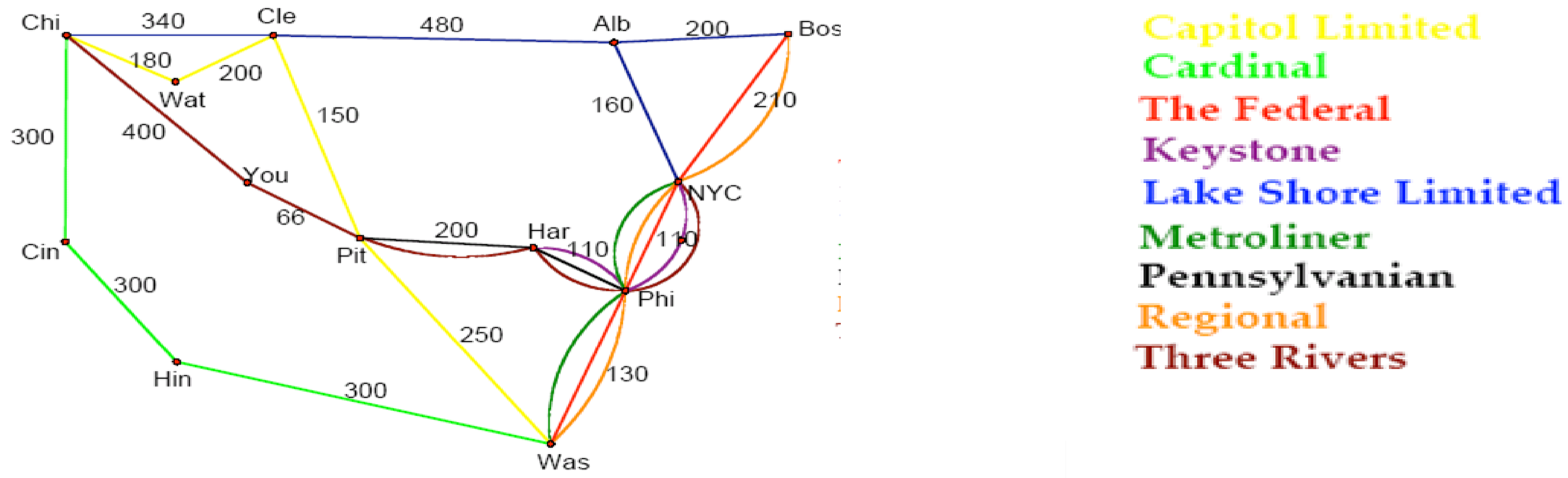
a. What are the major cities in the model? What makes a city important?
b. Are there some trains that are more important then others? How do you know that they are more important?
c. As consultants what are some (at least three) ways we could help Amtrak save money?

One problem with removing edges is deciding how many edges to remove and which ones they should be. A minimum spanning tree will ensure that all cities are still connected to. When creating such a tree, there are many factors that may be accounted for:

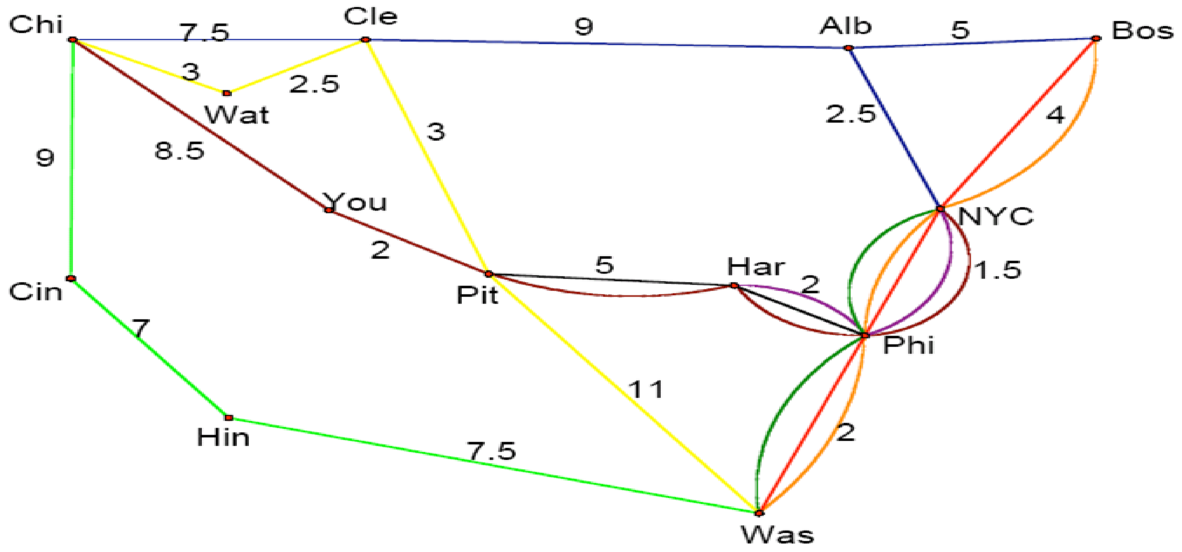
- Traffic on each train – Could be a factor of whether the route remains or to be removed.
- Cost of track-depending on the distance it covers. When there are multiple routes to a city, it may be cheaper to eliminate the longer tracks.
- Travel time/distance: it may be important to take into account the travel time between any two cities. It may be useful to calculate the maximum and average travel time/distance between all pairs of cities in the graph.

Examples of graphs, where the numbers on each segment represents:

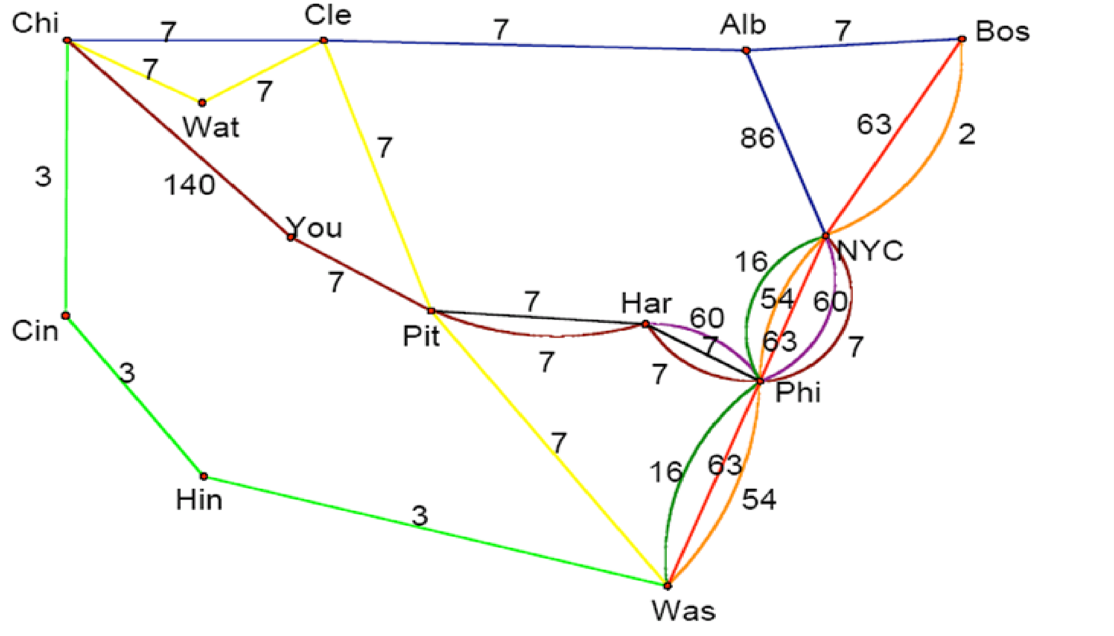
a) Distance



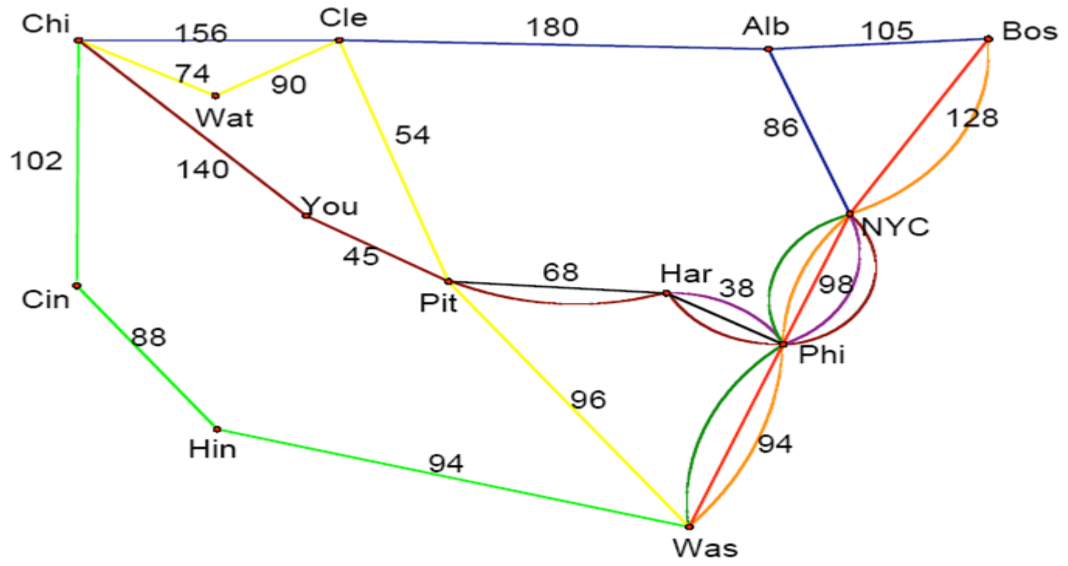
b) Time in hours



c) Frequency on each route



d) Cost in Euros



Algebra representation examples

a) Distance matrix showing shortest path between two routes

	Albany	Boston	New York	Philly	Wash DC	Pitts
Albany	0	144	138	216	342	532
Boston	144	0	282	360	486	676
New York	138	282	0	78	204	394
Philly	216	360	78	0	126	316
Wash DC	342	486	204	126	0	190
Pitts	532	676	394	316	190	0

b) Matrix showing If the cities are connected

	Albany	Boston	New York	Philly	Wash DC	Pitts
Albany	1	1	1	1	0	0
Boston	1	1	1	1	0	0
New York	1	1	3	1	1	1
Philly	1	1	1	4	1	1
Wash DC	0	0	1	1	2	1
Pitts	0	0	1	1	1	1

c) Fares for shortest route and so on.

Customer’s Point of View

Major points of focus:

- Travel Time
- Cost
- Number of Train Changes
- Availability of Service

Now that the students have an understanding of what the graph looks like and an understanding of what the problem is and why it is important they are ready for the next part.

A local problem can be given, or they can use the one given.

Problem

Below is a list of all trains servicing the northeast, as well as the connecting cities.

Acela Express: Boston → New York → Philadelphia → Washington
Capital Limited: Pittsburgh → Cleveland → Waterloo → Chicago
Cardinal: Washington → Hinton → Cincinnati → Lafayette → Chicago
The Federal: Boston → New York → Philadelphia → Washington
Keystone: New York → Philadelphia → Harrisburg
Lake Shore Limited: Chicago → Cleveland → Albany → Boston or New York
Metroliner: New York → Philadelphia → Washington
Pennsylvanian: Pittsburgh → Philadelphia → New York
Three Rivers: New York → Youngstown → Chicago

Number of miles between any two cities

	Albany	Boston	New York	Philly	Wash DC	Pitts	Cleve.	Chicago	Cincy	Harris
Albany	0	144	138	216	342	532	646	958	1208	310
Boston	144	0	282	360	486	676	790	1102	1352	376
New York	138	282	0	78	204	394	508	820	1070	172
Philly	216	360	78	0	126	316	430	742	992	94
Wash DC	342	486	204	126	0	190	304	616	866	220
Pitts	532	676	394	316	190	0	114	426	676	410
Cleveland	646	790	508	430	304	114	0	312	562	524
Chicago	958	1102	820	742	616	426	312	0	250	836
Cincy	1203	1352	1070	992	866	676	562	250	0	1086
Harris	310	376	172	94	220	410	524	836	1086	0

Students

a) Connect the cities using graphs

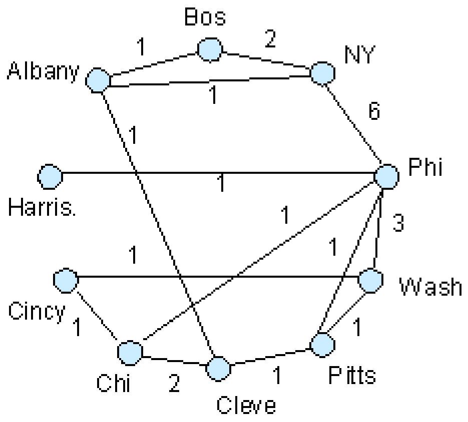
b) Chose a path that can optimized the trains routes for connected cities.

c) Identify the most central cities based on their distances and place them in rank position.
(hint the city with the smallest distance to all the rest will be the most central)

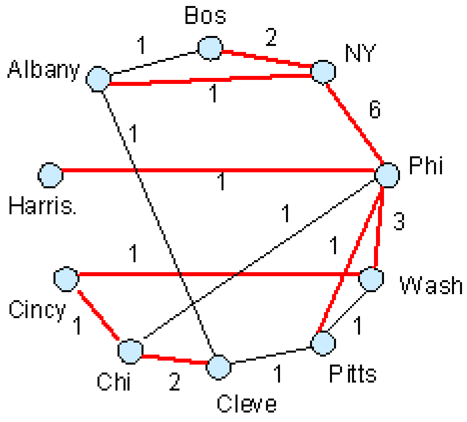
d) Chose a path that can optimized the trains routes with respect of their distances.

For the Teacher-solutions

a)



b)



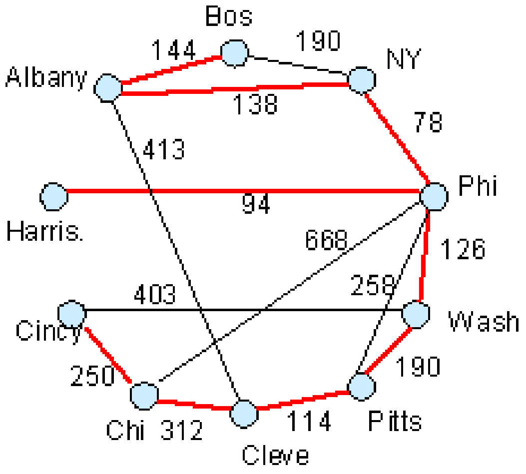
This optimization path is open to discussion, though the tracks with the highest weights have been kept while only tracks with a weight of 1 have been dropped. What's important is that no city has become disconnected from the network.

To find the most central city, find the row sum (or column since it is symmetric) for each city. Finally, rank the cities to figure out which is the most centrally located and which was the most isolated:

c)

	Albany	Boston	New York	Philly	Wash DC	Pitts	Cleve.	Chicago	Cincy	Harris
Albany	0	144	138	216	342	532	646	958	1208	310
Boston	144	0	282	360	486	676	790	1102	1352	376
New York	138	282	0	78	204	394	508	820	1070	172
Philly	216	360	78	0	126	316	430	742	992	94
Wash DC	342	486	204	126	0	190	304	616	866	220
Pitts	532	676	394	316	190	0	114	426	676	410
Cleveland	646	790	508	430	304	114	0	312	562	524
Chicago	958	1102	820	742	616	426	312	0	250	836
Cincy	1203	1352	1070	992	866	676	562	250	0	1086
Harris	310	376	172	94	220	410	524	836	1086	0
TOT	**4489**	**5568**	**3666**	**3354**	**3354**	**3734**	**4190**	**6062**	**8062**	**4028**
Rank	6	7	2	1	1	3	5	8	9	4

d) Since the idea is to use the least amount of track as possible, one option is to create a minimal spanning tree.



Since the idea is to use the least amount of track as possible, one option is to create a minimal spanning tree. There is now 1446 miles of track compared to 3378 miles of original track: