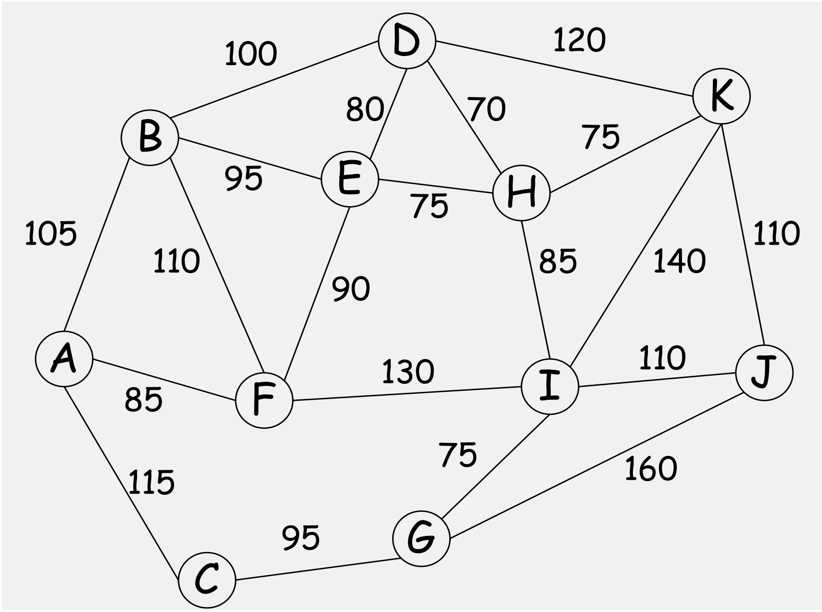
IEOR151

Lab 10: Solving Simple TSP in Excel

Long He
Dept of Industrial Engineering & Operations Research
Fall 2013

Traveling Salesman Problem (TSP)

- Objective
  - To find a minimum-length route that begins at some node-it could be any node in the network-and ends at the same node, visiting each node exactly once.



Heuristic Algorithms

- Minimum spanning tree based heuristic
- Nearest insertion heuristic
- Christofides' heuristic

A Minimum Spanning Tree Based Heuristic

- *Step 1*: Construct a minimum spanning tree
- *Step 2*: Let the *root* be an arbitrary vertex
- *Step 3*: Traverse all the vertices by depth-first search, record the sequence of vertices (both visited and unvisited)
- *Step 4*: Use shortcut strategy to generate a feasible tour

Worst-case Analysis

Let L^{MST} denotes the length of the tour generated by above strategy, then we have

$$L^{MST} \leq 2W^* \leq 2L^*$$

Where W^* denotes the length of the minimum spanning tree.

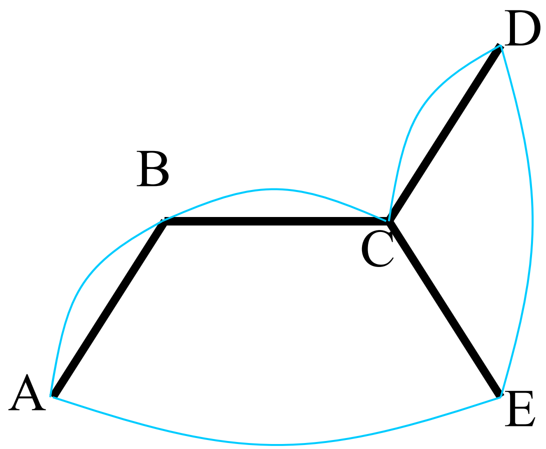
And this bound is tight.

Nearest Insertion Heuristic

- *Step 1*: Choose an arbitrary node v and let the cycle C consist of only v.
- *Step 2*: Find a node outside C closest to a node in C; call it k.
- *Step 3*: Find an edge $\{i, j\}$ in C such that $d_{ik}+d_{kj}-d_{ij}$ is minimal.
- *Step 4*: Construct a new cycle C by replacing $\{i, j\}$ with $\{i, k\}$ and $\{k, j\}$.
- *Step 5*: If the current cycle C contains all the vertices, stop. Otherwise, go to *Step 2*.

Worst Case Analysis

The length of current tour *C* is no more than the twice of the dark line length.



The dark line is nothing, but a minimum spanning tree (exactly same as Prim's Algorithm), thus

$$L^{MST} \leq 2W^* \leq 2L^*$$

Christofides' Heuristic

1. Find the minimum spanning tree
2. Find the odd-degree nodes of the MST
3. Find the optimal matching of the odd-degree nodes and add links of the matching to the MST
4. Draw an Euler tour on the resulting graph

Step 3

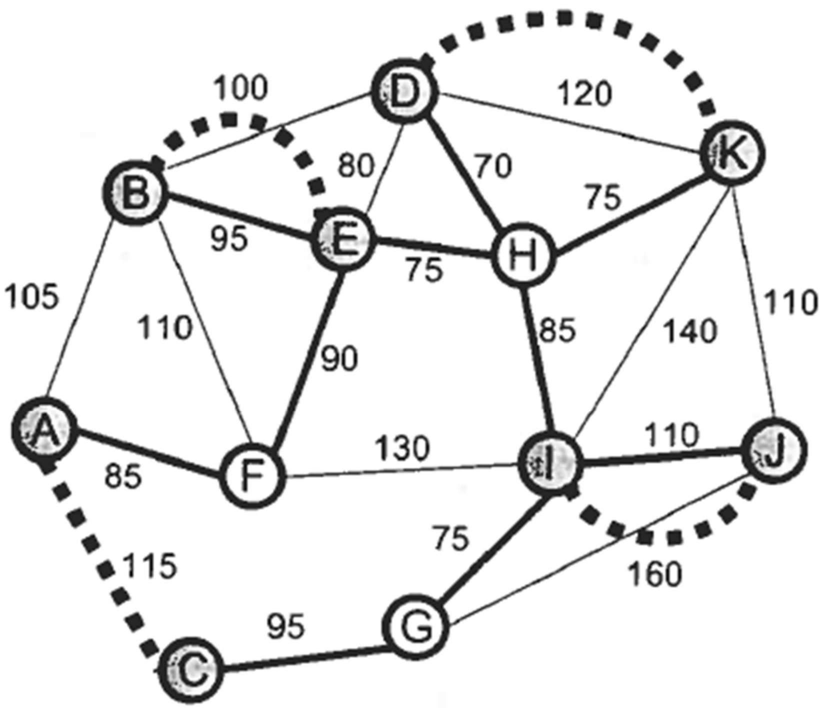


Figure 10.22. MST with optimal matching of odd degree (shaded) nodes

Step 4

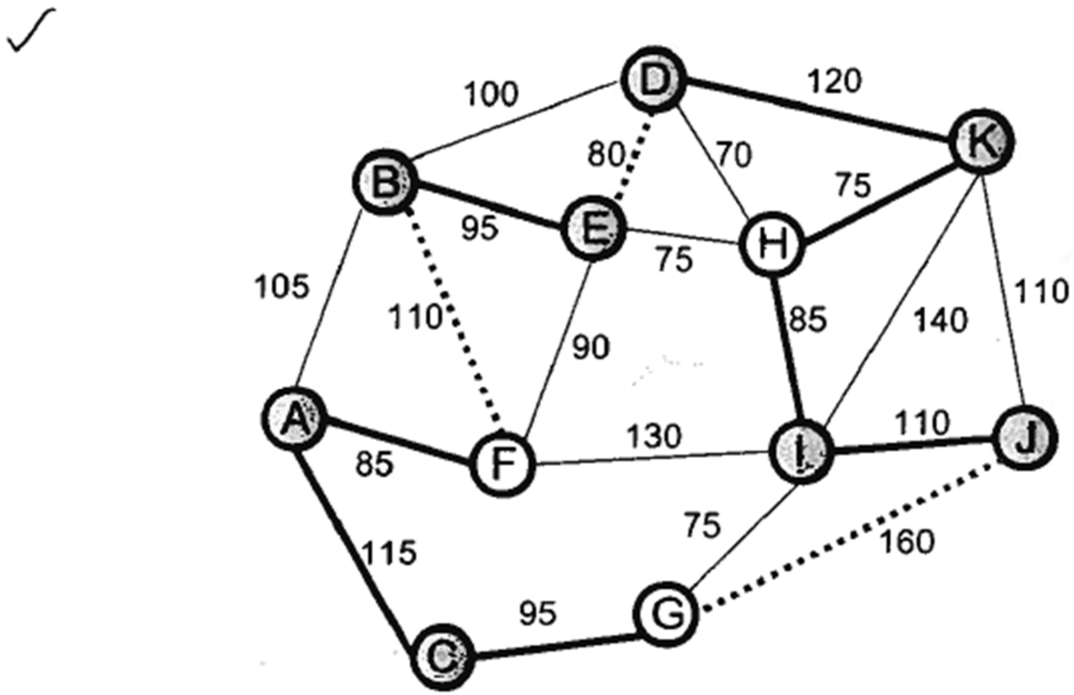


Figure 10.23. Christofides' heuristic result after removing multiple visits to nodes E, H, and I

Optimal Matching -- Example

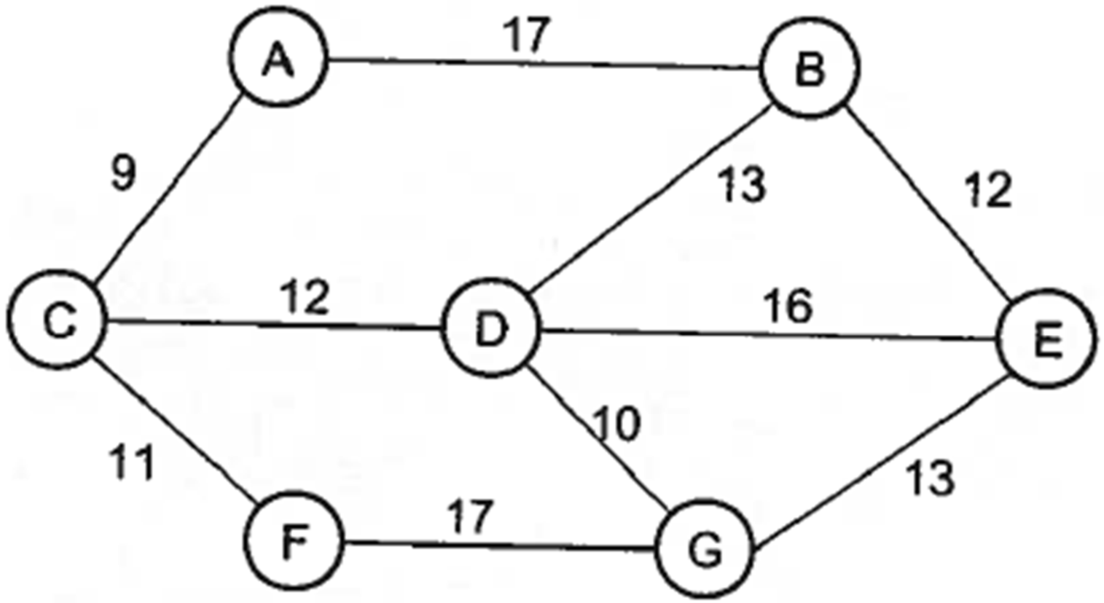


Figure 10.3. Example undirected network

Example

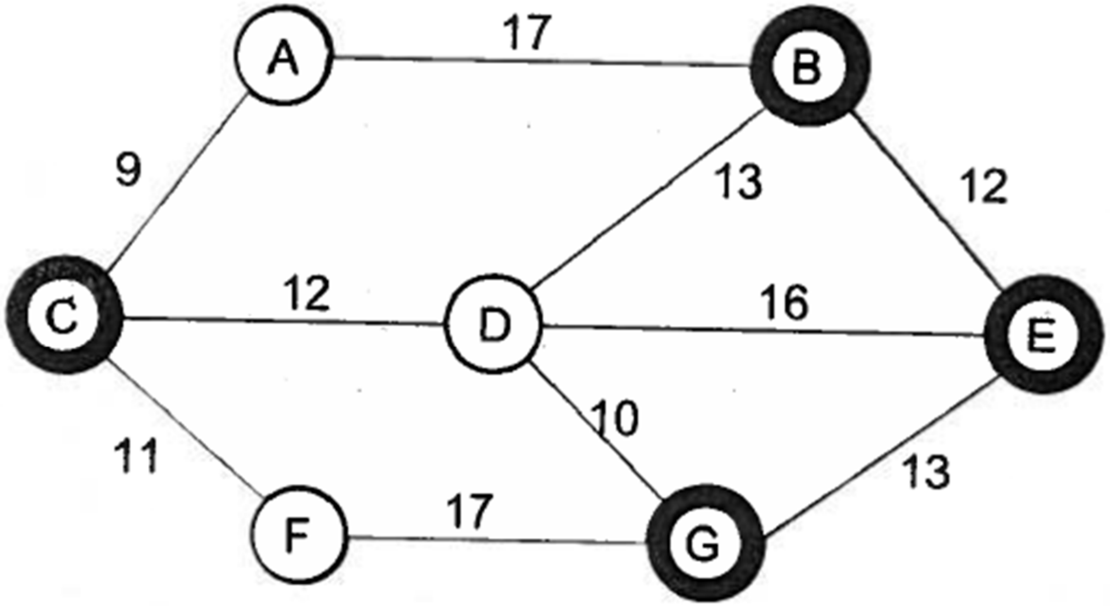


Figure 10.4. Simple network of Figure 10.3 with odd-degree nodes highlighted

Example

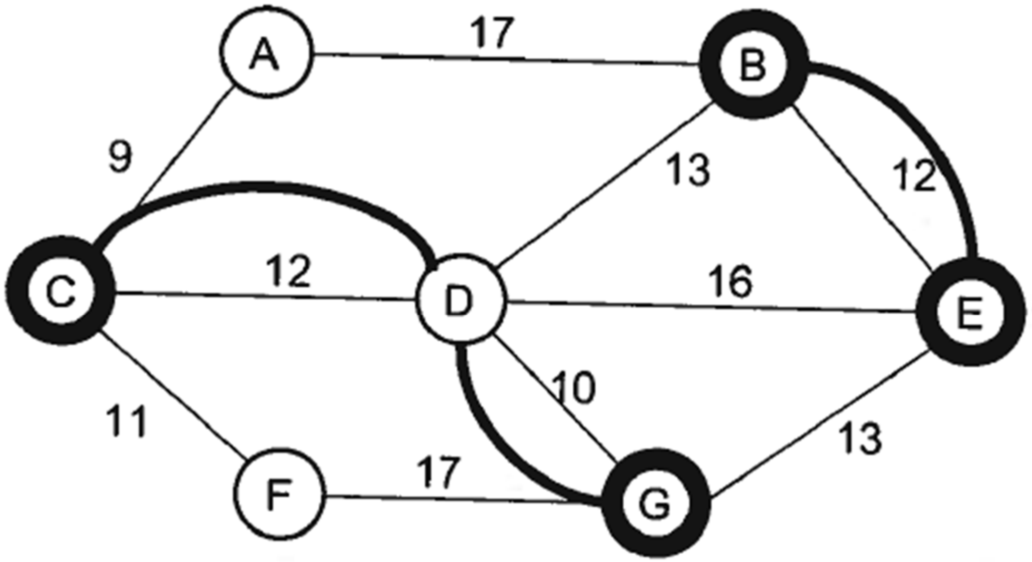


Figure 10.5. Optimal matching of the odd-degree nodes of Figure 10.4

Finding Optimal Matching

TABLE 10.1. Pairs of odd-degree nodes than can be paired from the network of Figure 10.4

Pair Nodes	Repeat Links	Total Distance
B, C	BD, DC	25
✓ B, E	BE	12
B, G	BD, DG	23
C, E	CD, DE	28
✓ C, G	CD, DG	22
E, G	EG	13

TSP Formulation

- Inputs and sets
  - N= the set of nodes in the network
  - d_{jk}= the distance between nodes j and k (to avoid self-loops, set $d_{jj}=\infty$ for all j in N)
- Decision variables
  - X_{jk}= 1 if the tour goes directly from j to k;
    0 otherwise

TSP Formulation

$$Min \quad \sum_{j \in N} \sum_{k \in N} d_{jk} X_{jk}$$

$$s.t. \quad \sum_{k \in N} X_{jk} = 1 \qquad \forall j \in N$$

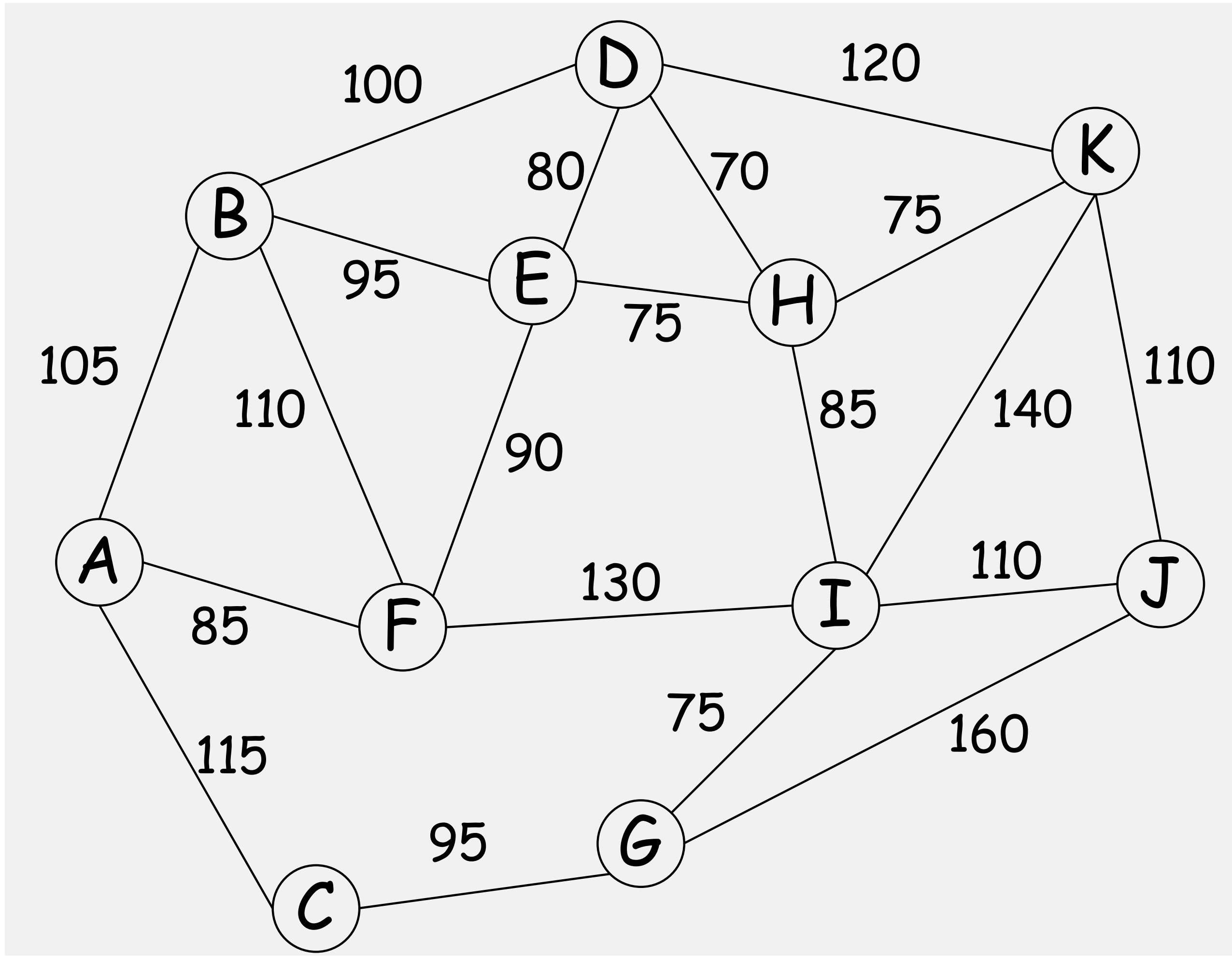
$$\sum_{j \in N} X_{jk} = 1 \qquad \forall k \in N$$

Subtour Elimination Constraints

$$\sum_{j \in S} \sum_{k \in S} X_{jk} \leq |S| - 1 \qquad \forall S \subset N, 2 \leq |S| \leq |N| - 1$$

$$X_{jk} = \{0,1\} \qquad \forall\, j \in N; k \in N$$

Example

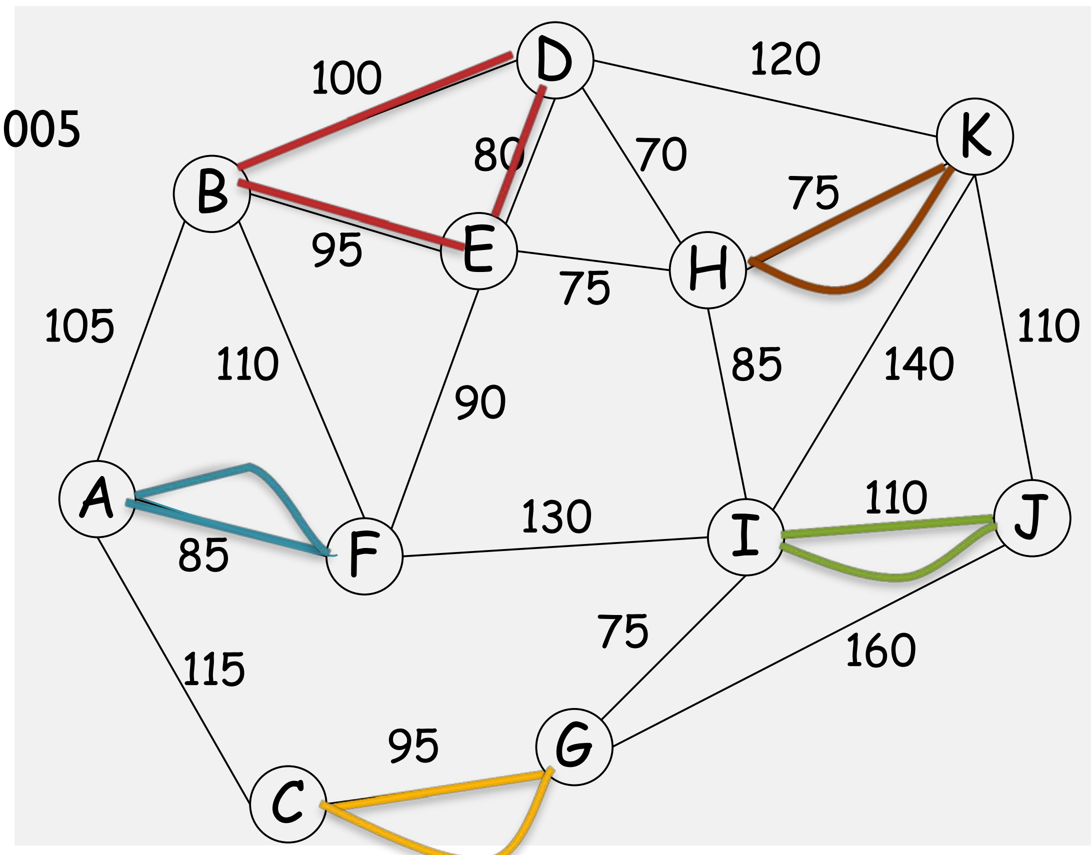


D
100
120
K
B
80
70
75
95
E
H
75
105
110
110
85
140
90
A
110
J
130
I
85
F
75
160
115
95
G
C

Example

- Solving without "subtour elimination constraints"

Total distance= 1005



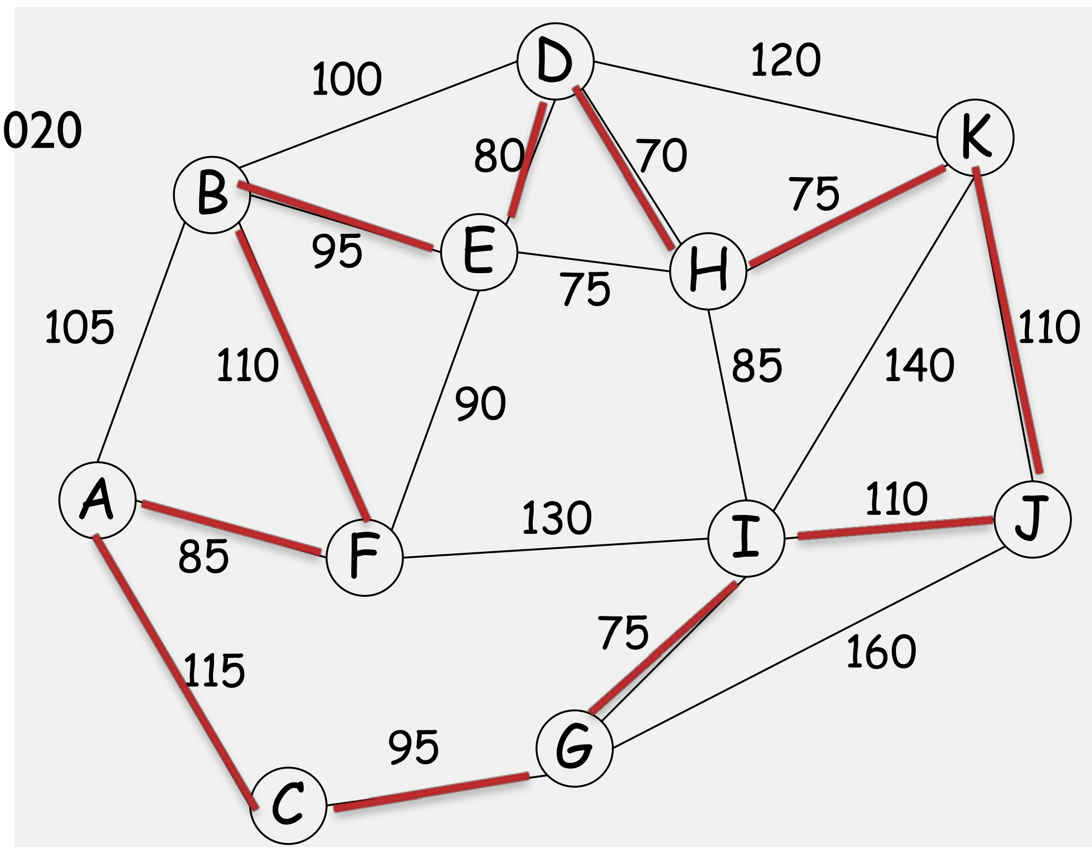
Example

- Luckily, we don't need $2^{11}-11-2= 2035$ "subtour elimination constraints" for this example.
- Let's eliminate the existing 6 subtours. Namely: A-F-A, B-D-E-B, B-E-D-B, C-G-C, H-K-H, I-J-I.

Example

- Solution

Total distance= 1020



Exercise

- Solve the example in Excel Solver.
- Show me the result and take attendance

Hint: add subtour elimination constraints below

$$A-F-A \quad X_{AF}+X_{FA}\le 1$$

$$B-D-E-B \quad X_{BD}+X_{DE}+X_{EB}\le 2$$

$$B-E-D-B \quad X_{BE}+X_{ED}+X_{DB}\le 2$$

$$C-G-C \quad X_{CG}+X_{GC}\le 1$$

$$H-K-H \quad X_{HK}+X_{KH}\le 1$$

$$I-J-I \quad X_{IJ}+X_{JI}\le 1$$