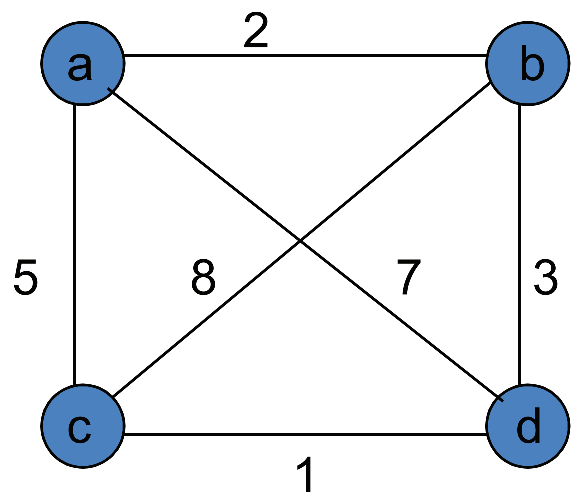
Branch and Bound

Traveling salesman problem

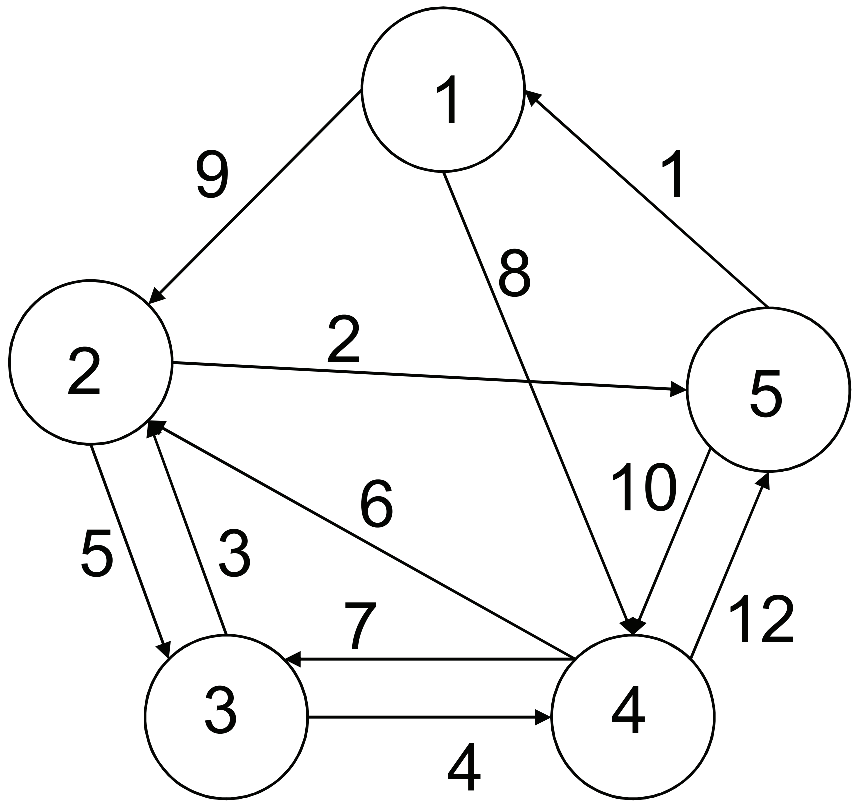


{ a, b, c, d }
represents 4 cities

The weights
represent distances
between cities

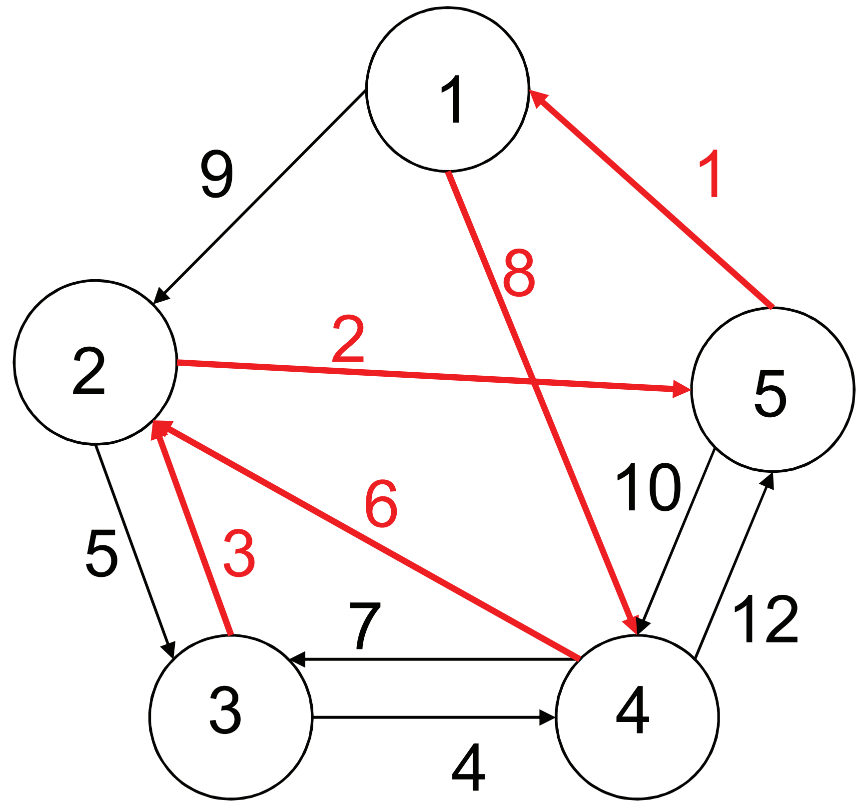
Problem: find the shortest path from a city (say a), visit all other cities exactly once, and return to the city where it started (city a).

Bound on TSP Tour



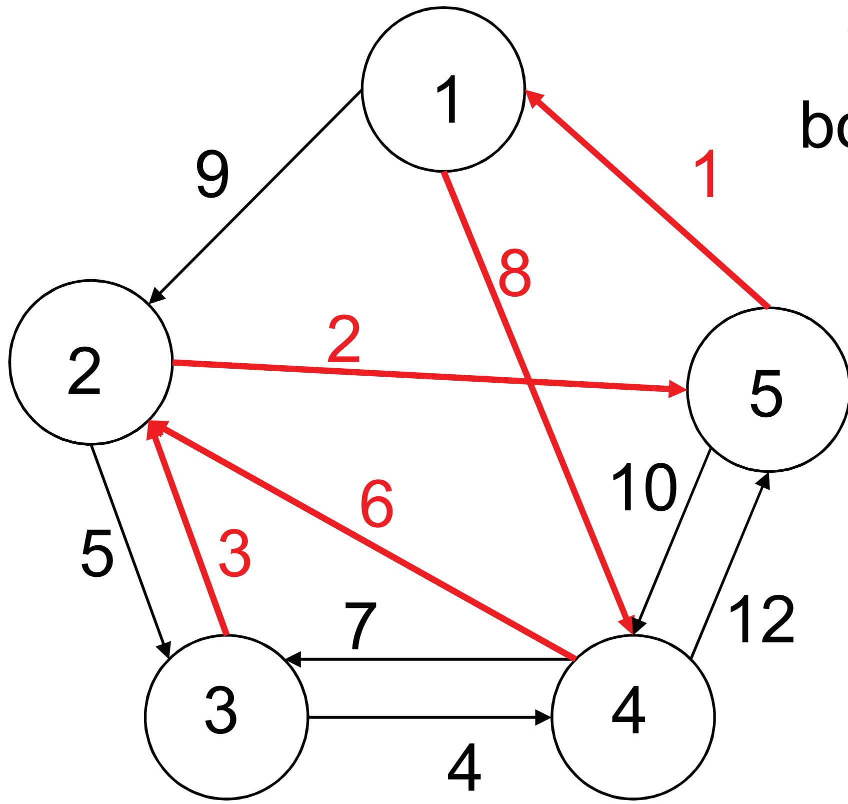
Every tour must leave every vertex and
and arrive at every vertex.

Bound on TSP Tour



What's the cheapest way to leave each vertex?

Bound on TSP Tour



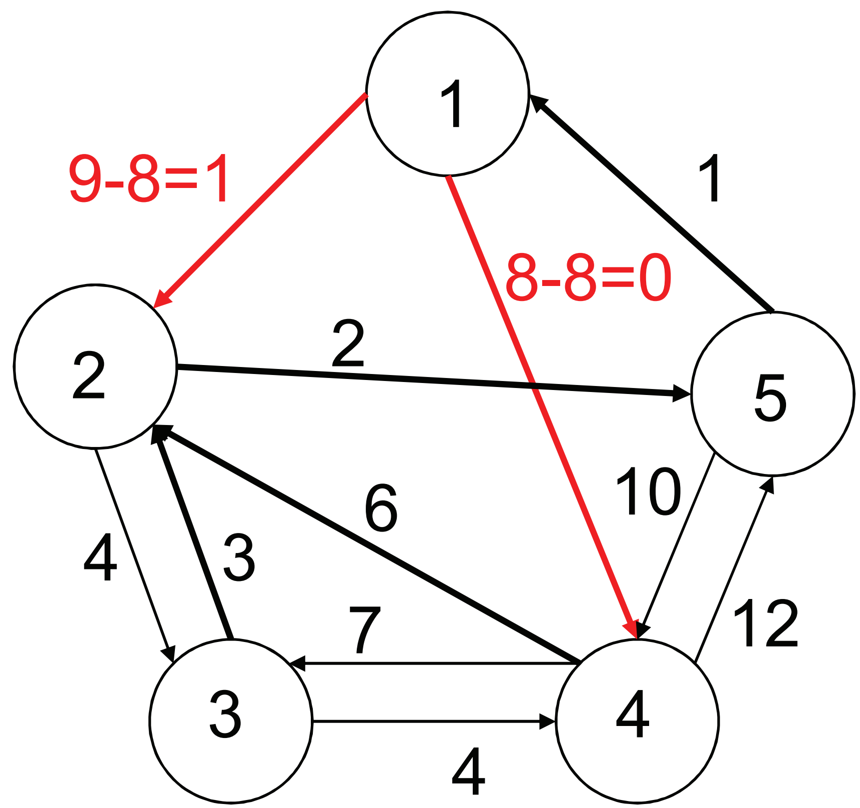
rough draft

bound=8+6+3+2+1

=20

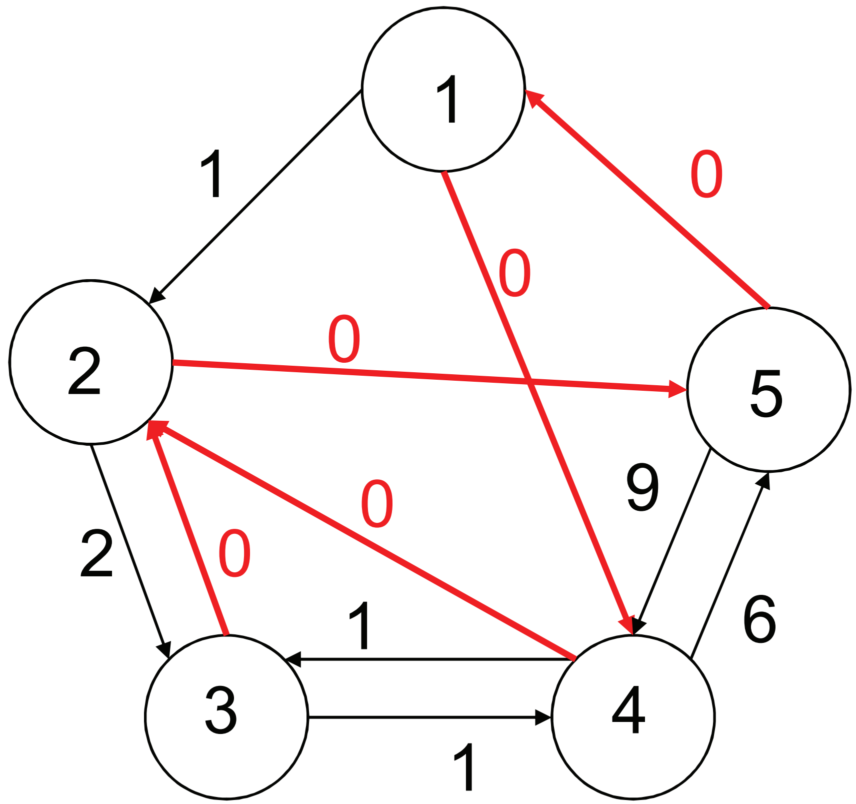
Save the sum of those costs in the bound (as a rough draft).
Can we find a tighter lower bound?

Bound on TSP Tour



For a given vertex, subtract the least cost departure from each edge leaving that vertex.

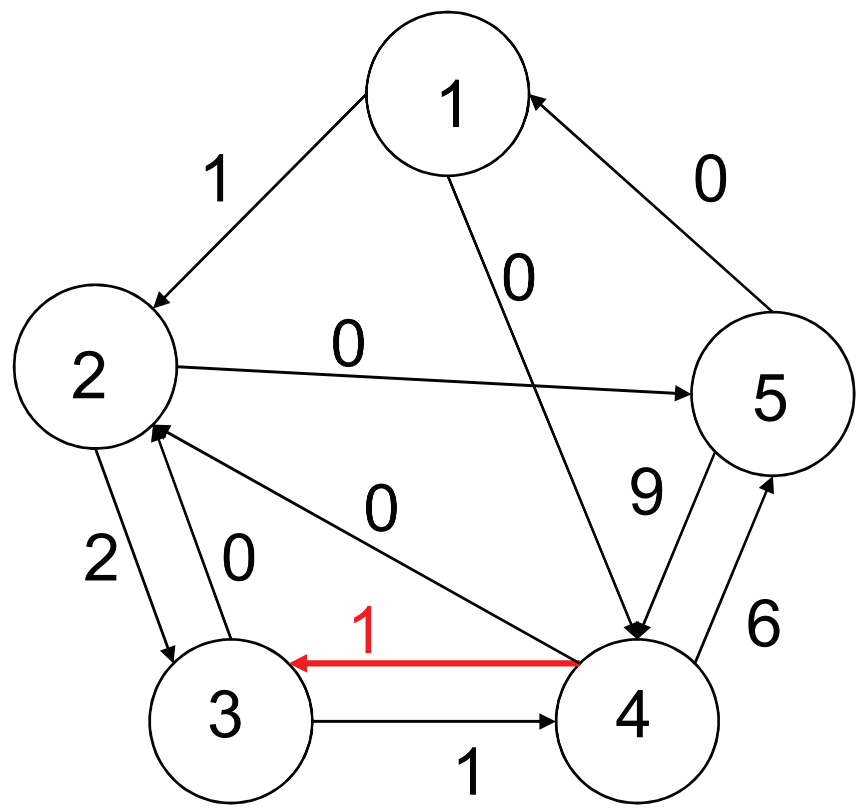
Bound on TSP Tour



bound=20

Repeat for the other vertices.

Bound on TSP Tour

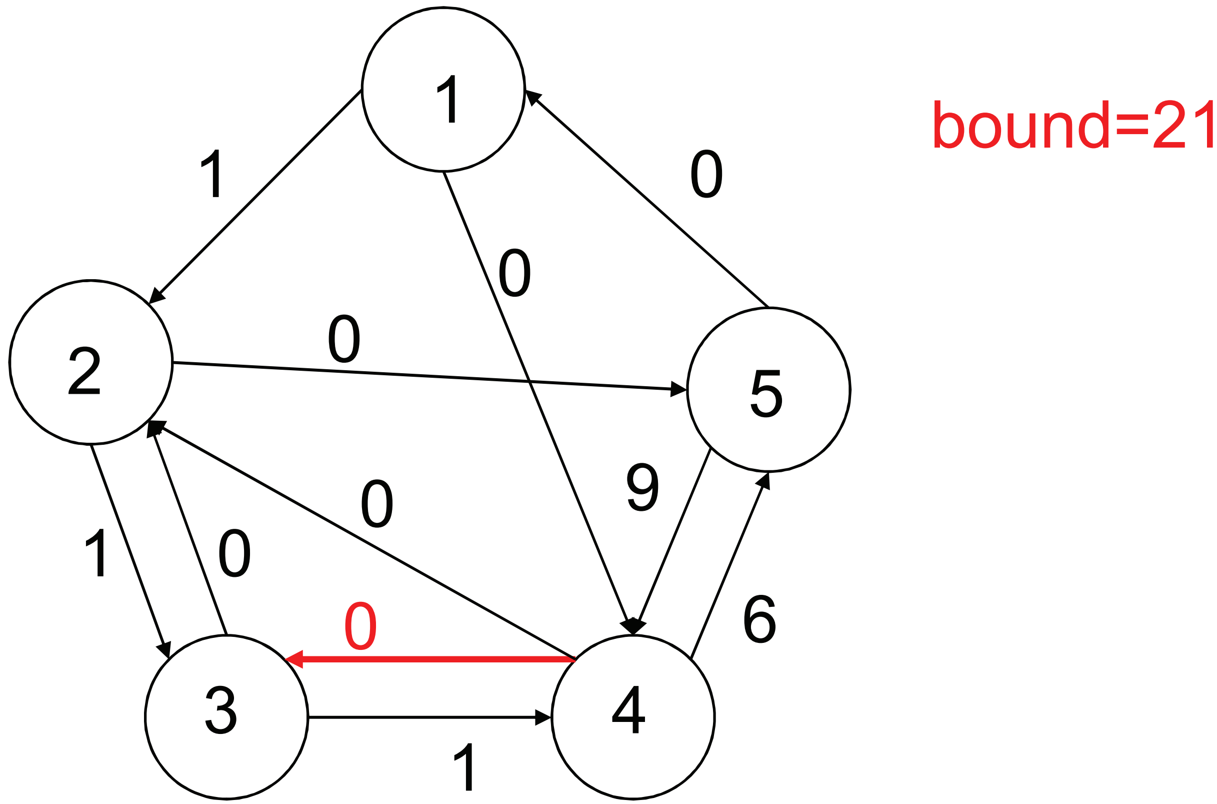


bound=20

Does that set of edges now having 0 residual cost arrive at every vertex?

In this case, the edges never arrive at vertex 3.

Bound on TSP Tour



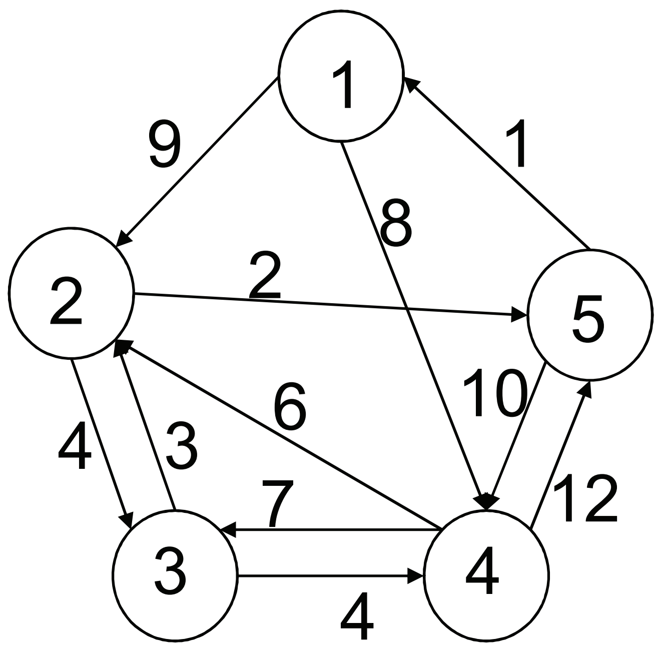
We have to take an edge to vertex 3 from somewhere. Assume we take the cheapest. Subtract its cost from other edges entering vertex 3 and add the cost to the bound.

We have just tightened the bound.

The Bound

- It will cost at least this much to visit all the vertices in the graph.
  - there's no cheaper way to get in and out of each vertex.
  - the edges are now labeled with the *extra* cost of choosing another edge.

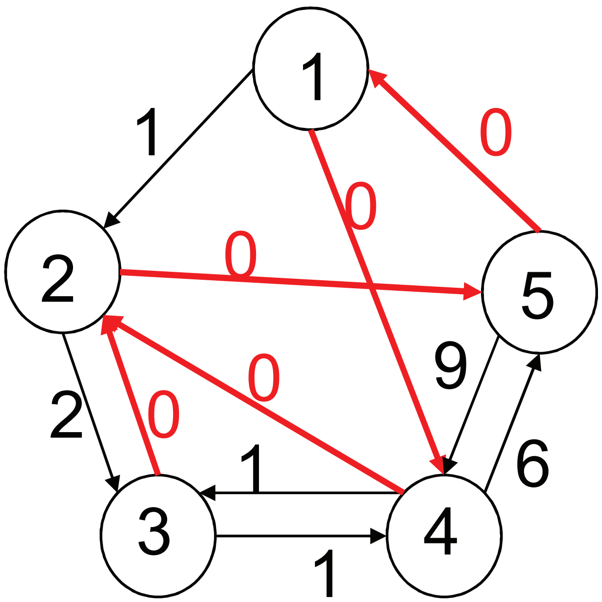
Bound on TSP Tour



$$\begin{pmatrix} 999 & 9 & 999 & 8 & 999 \\ 999 & 999 & 4 & 999 & 2 \\ 999 & 3 & 999 & 4 & 999 \\ 999 & 6 & 7 & 999 & 12 \\ 1 & 999 & 999 & 10 & 999 \end{pmatrix}$$

Algorithms do this using a cost matrix.

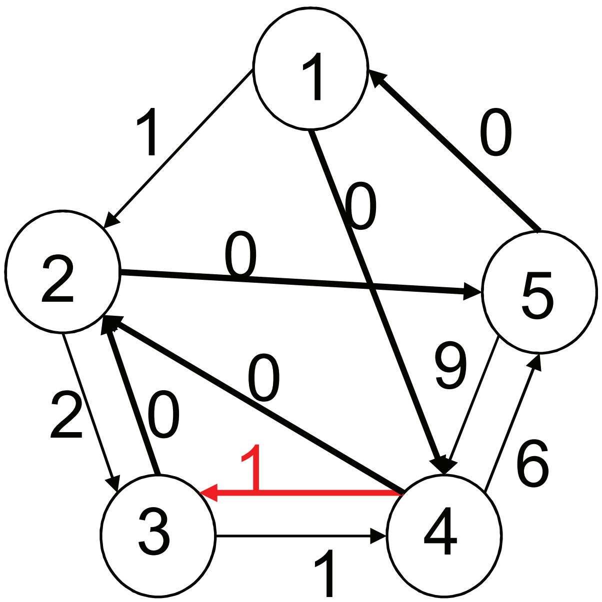
Bound on TSP Tour



$$
\begin{array}{c c}
\begin{pmatrix}
999 & 1 & 999 & 0 & 999 \\
999 & 999 & 2 & 999 & 0 \\
999 & 0 & 999 & 1 & 999 \\
999 & 0 & 1 & 999 & 6 \\
0 & 999 & 999 & 9 & 999
\end{pmatrix}
&
\begin{array}{c}
8 \\ 2 \\ 3 \\ 6 \\ 1 \\ \hline 20
\end{array}
\end{array}
$$

Reduce all rows.

Bound on TSP Tour



$$\begin{pmatrix} 999 & 1 & 999 & 0 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 1 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{pmatrix}$$

bound: 20 + 1 = 21

Then reduce column #3. Now we have a tight bound.

Using this bound for TSP in B&B

start at ~~node~~ vertex 1 in graph (arbitrary)

$$\begin{bmatrix} 999 & 1 & 999 & 0 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 1 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{bmatrix}$$

bound = 21

1to2

$$\begin{bmatrix} 999 & 1 & 999 & 0 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 1 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{bmatrix}$$

bound = 21+1

1to3

$$\begin{bmatrix} 999 & 1 & 999 & 0 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 1 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{bmatrix}$$

infeasible

1to4

$$\begin{bmatrix} 999 & 1 & 999 & 0 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 1 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{bmatrix}$$

bound = 21

1to5

$$\begin{bmatrix} 999 & 1 & 999 & 0 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 1 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{bmatrix}$$

infeasible

Using this bound for TSP in B&B

start at ~~node~~ vertex 1 in graph (arbitrary)

$$\begin{bmatrix} 999 & 1 & 999 & 0 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 1 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{bmatrix}$$

bound = 21

1to2

$$\begin{bmatrix} 999 & \boxed{1} & 999 & 0 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 1 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{bmatrix}$$

bound = 21+1

1to3

$$\begin{bmatrix} 999 & 1 & 999 & 0 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 1 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{bmatrix}$$

infeasible

1to4

$$\begin{bmatrix} 999 & 1 & 999 & 0 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 1 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{bmatrix}$$

bound = 21

1to5

$$\begin{bmatrix} 999 & 1 & 999 & 0 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 1 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{bmatrix}$$

infeasible

Focus: going from 1 to 2

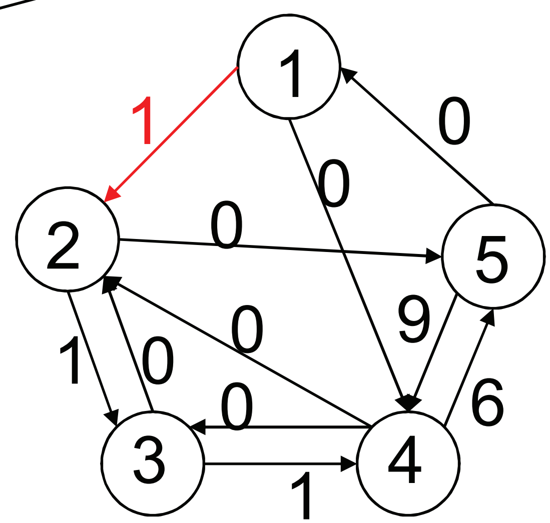
$$\begin{bmatrix} 999 & 1 & 999 & 0 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 1 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{bmatrix}$$

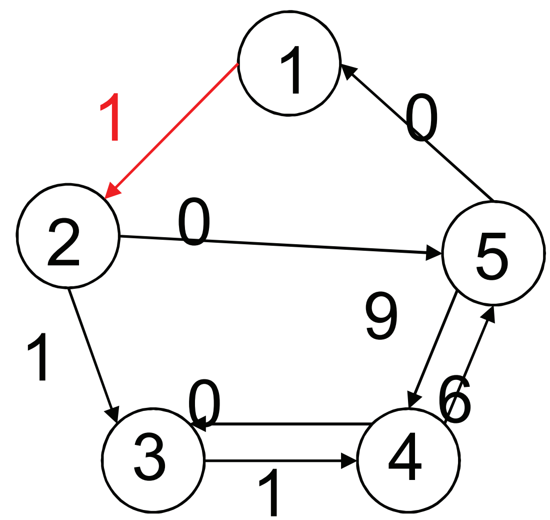
bound = 21

1to2

$$\begin{bmatrix} 999 & 999 & 999 & 999 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 999 & 999 & 1 & 999 \\ 999 & 999 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{bmatrix}$$

bound = 21+1





Add extra cost from 1 to 2, exclude edges from 1 or into 2.

Focus: going from 1 to 2

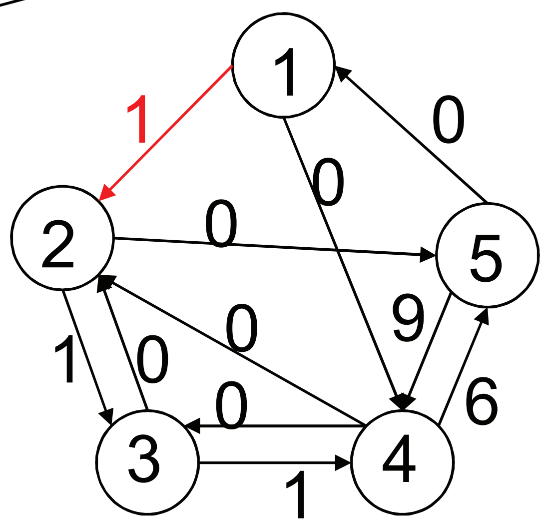
$$\begin{pmatrix} 999 & 1 & 999 & 0 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 1 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{pmatrix}$$

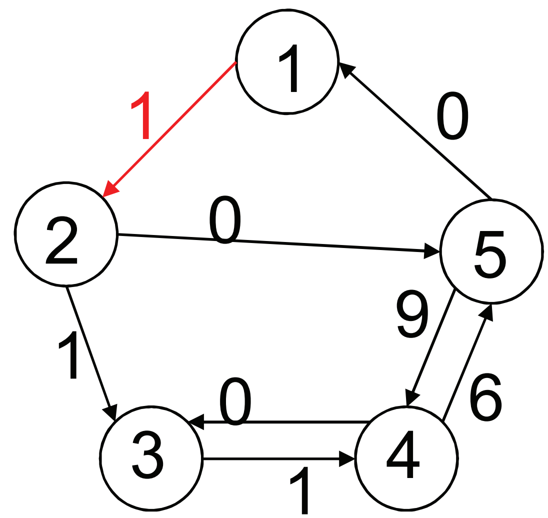
bound = 21

1to2

$$\begin{pmatrix} 999 & 999 & 999 & 999 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 999 & 999 & 1 & 999 \\ 999 & 999 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{pmatrix}$$

bound = 21+1+1





No edges into vertex 4 w/ 0 reduced cost.

Focus: going from 1 to 2

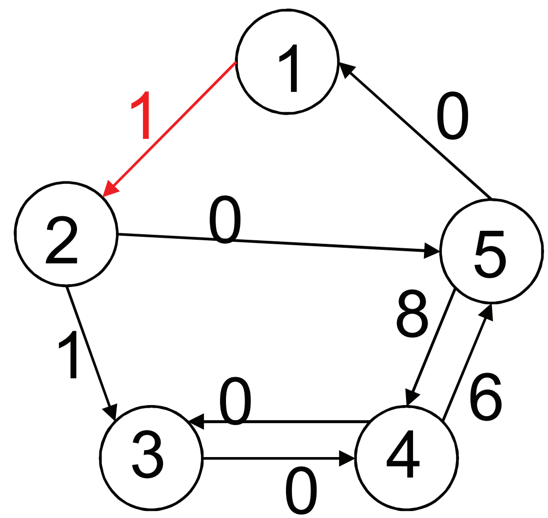
$$\begin{bmatrix} 999 & 1 & 999 & 0 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 1 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{bmatrix}$$

bound = 21

1to2

$$\begin{bmatrix} 999 & 999 & 999 & 999 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 999 & 999 & 0 & 999 \\ 999 & 999 & 0 & 999 & 6 \\ 0 & 999 & 999 & 8 & 999 \end{bmatrix}$$

bound = 21+1+1 = 23



Add cost of reducing edge into vertex 4.

Bounds for other choices.

1to2(23),1to4(21)

$$\begin{pmatrix} 999 & 1 & 999 & 0 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 1 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{pmatrix}$$

bound = 21

1to2

$$\begin{pmatrix} 999 & 999 & 999 & 999 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 999 & 999 & 0 & 999 \\ 999 & 999 & 0 & 999 & 6 \\ 0 & 999 & 999 & 8 & 999 \end{pmatrix}$$

bound = 23

1to3

$$\begin{pmatrix} 999 & 999 & 999 & 0 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 1 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{pmatrix}$$

bound = 999

1to4

$$\begin{pmatrix} 999 & 999 & 999 & 999 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 999 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 999 & 999 \end{pmatrix}$$

bound = 21

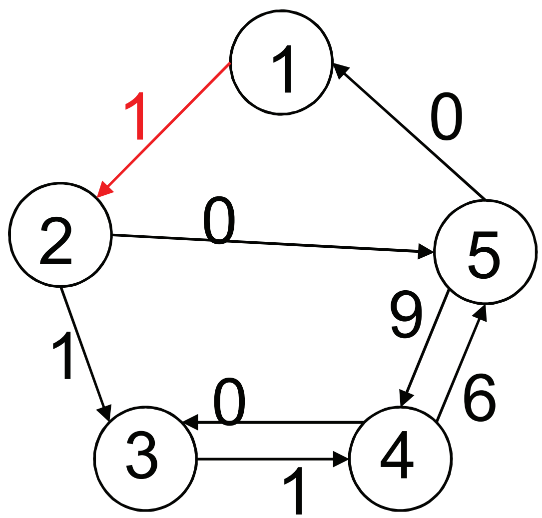
1to5

$$\begin{pmatrix} 999 & 1 & 999 & 0 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 1 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 9 & 999 \end{pmatrix}$$

bound = 999

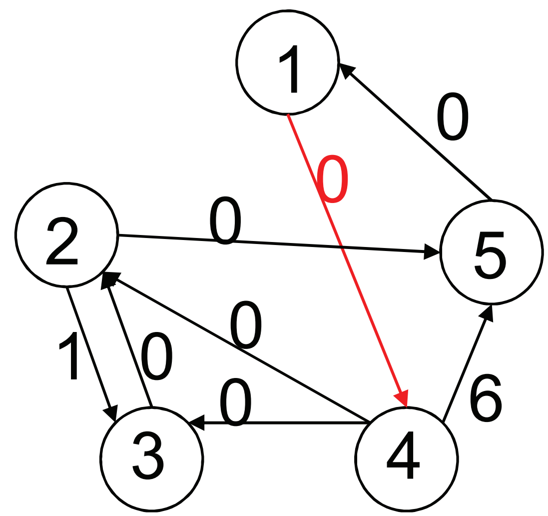
Leaves us with Two Possibilities on Priority Queue

1 -> 2



bound = 23

1 -> 4



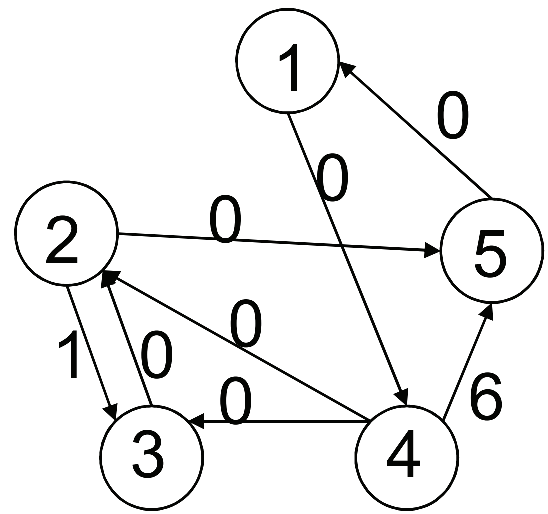
bound = 21

Leaving Vertex 4

4to2(22), 4to3(21)
4to5(28),1to2(23),

$$\begin{pmatrix} 999 & 999 & 999 & 999 & 999 \\ 999 & 999 & 1 & 999 & 0 \\ 999 & 0 & 999 & 999 & 999 \\ 999 & 0 & 0 & 999 & 6 \\ 0 & 999 & 999 & 999 & 999 \end{pmatrix}$$

bound = 21



4to2

$$\begin{pmatrix} 999 & 999 & 999 & 999 & 999 \\ 999 & 999 & 0 & 999 & 0 \\ 999 & 999 & 999 & 999 & 999 \\ 999 & 999 & 999 & 999 & 999 \\ 0 & 999 & 999 & 999 & 999 \end{pmatrix}$$

bound = 22

4to3

$$\begin{pmatrix} 999 & 999 & 999 & 999 & 999 \\ 999 & 999 & 999 & 999 & 0 \\ 999 & 0 & 999 & 999 & 999 \\ 999 & 999 & 999 & 999 & 999 \\ 0 & 999 & 999 & 999 & 999 \end{pmatrix}$$

bound = 21

4to5

$$\begin{pmatrix} 999 & 999 & 999 & 999 & 999 \\ 999 & 999 & 0 & 999 & 999 \\ 999 & 0 & 999 & 999 & 999 \\ 999 & 999 & 999 & 999 & 999 \\ 0 & 999 & 999 & 999 & 999 \end{pmatrix}$$

bound = 28

Leaving Vertex 3

4to2(22), 3to2(21)
1to2(23),

$$\begin{pmatrix} 999 & 999 & 999 & 999 & 999 \\ 999 & 999 & 999 & 999 & 0 \\ 999 & 0 & 999 & 999 & 999 \\ 999 & 999 & 999 & 999 & 999 \\ 0 & 999 & 999 & 999 & 999 \end{pmatrix}$$

bound = 21

3to2

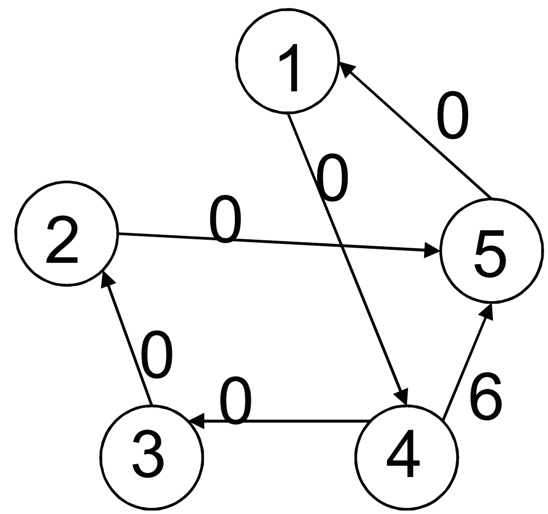
$$\begin{pmatrix} 999 & 999 & 999 & 999 & 999 \\ 999 & 999 & 999 & 999 & 0 \\ 999 & 999 & 999 & 999 & 999 \\ 999 & 999 & 999 & 999 & 999 \\ 0 & 999 & 999 & 999 & 999 \end{pmatrix}$$

bound = 21

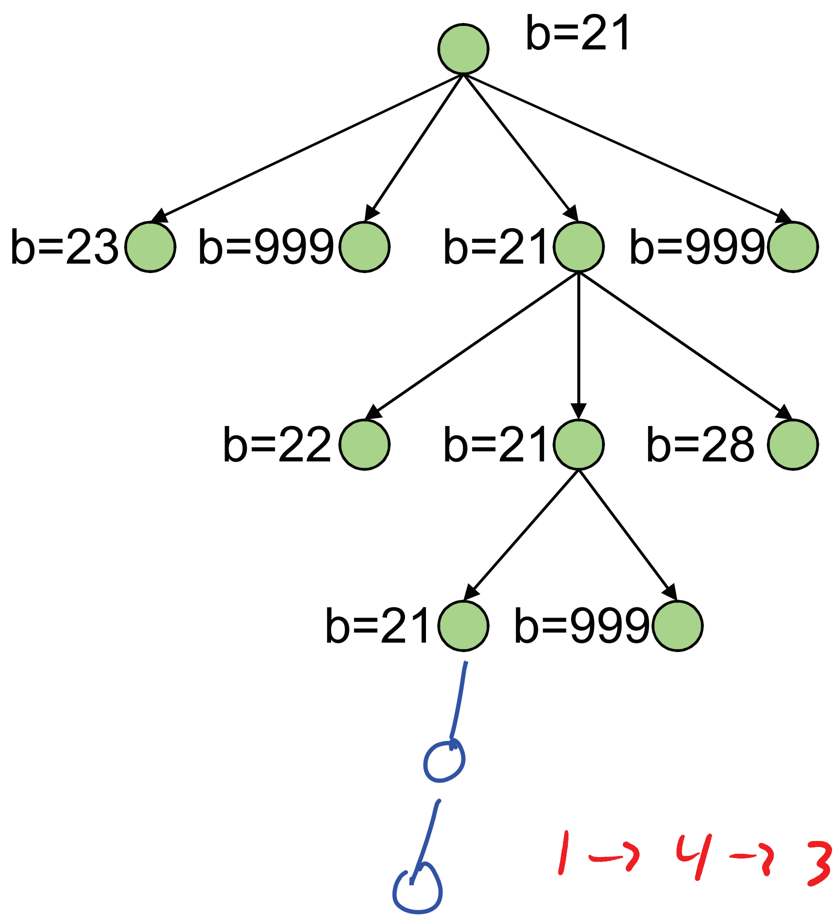
3to5

$$\cancel{\begin{pmatrix} 999 & 999 & 999 & 999 & 999 \\ 999 & 999 & 999 & 999 & 999 \\ 999 & 999 & 999 & 999 & 999 \\ 999 & 999 & 999 & 999 & 999 \\ 0 & 999 & 999 & 999 & 999 \end{pmatrix}}$$

bound = 999



Search Tree for This Problem



1 → 4 → 3 → 2 → 5 → 1