Edsger W. Dijkstra

Pioneer in the fields of Computer Science and Engineering

1972 received the Turing Award (like the Nobel prize for Computer Science)

Schlumberger Centennial Chair in Computing Sciences at the University of Texas at Austin, 1984-2000

Dijkstra, E. W. (1959) A note on two problems in connexion with graphs. *Numerische Mathematik*, 1. 269–271

Dijkstra Shortest Path Algorithm

Guaranteed to give the optimal solution

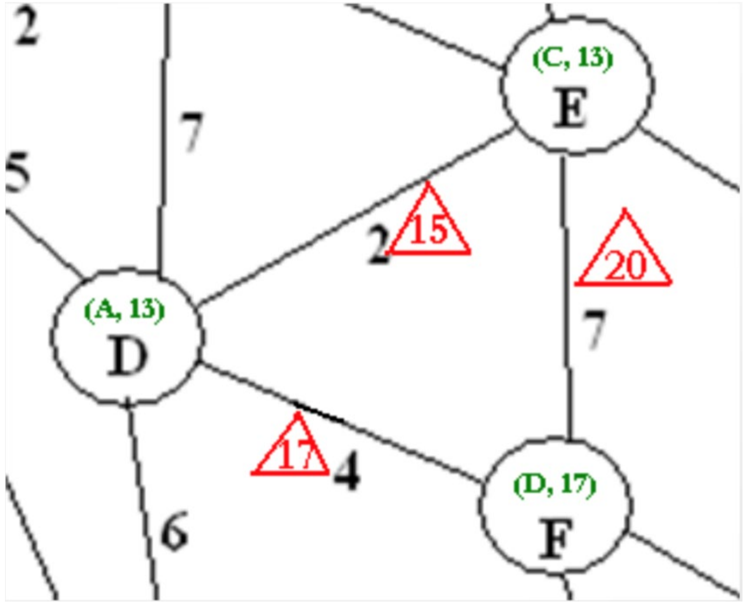
Label setting algorithm

Two types of labels

- **Temporary labels on Edges:**
  - **A representation of *total cost* to cross that edge *from the origin***
  - **Designated by a number surrounded by a triangle**

Permanent labels on Nodes:

- **Are two-part labels represented by an ordered pair in parentheses**
  - **The first part is the node from which you traveled to get to the node you are labeling**
  - **The second part is the total cost you have incurred to reach the node you are labeling from the origin**



Dijkstra’s Algorithm

1) Start at the origin vertex (S) and give it a permanent label

A) This permanent label will be(-,0) to represent coming from nowhere at zero cost

B) All other edges and vertices are unlabeled

2) For each permanently labeled vertex

A) Give each edge connected from it to an unlabeled vertex a temporary label denoting the total least cost to travel across that edge *from the origin*

B) Choose the edge with the smallest total cost (the smallest temporary label) and permanently label its connected vertex with the name of the vertex you traveled from and the cost to get there *from the origin*

C) Identify that edge as possibly being part of the final path solution (highlight the edge)

D) Eliminate temporary labels on edges between permanently labeled vertices

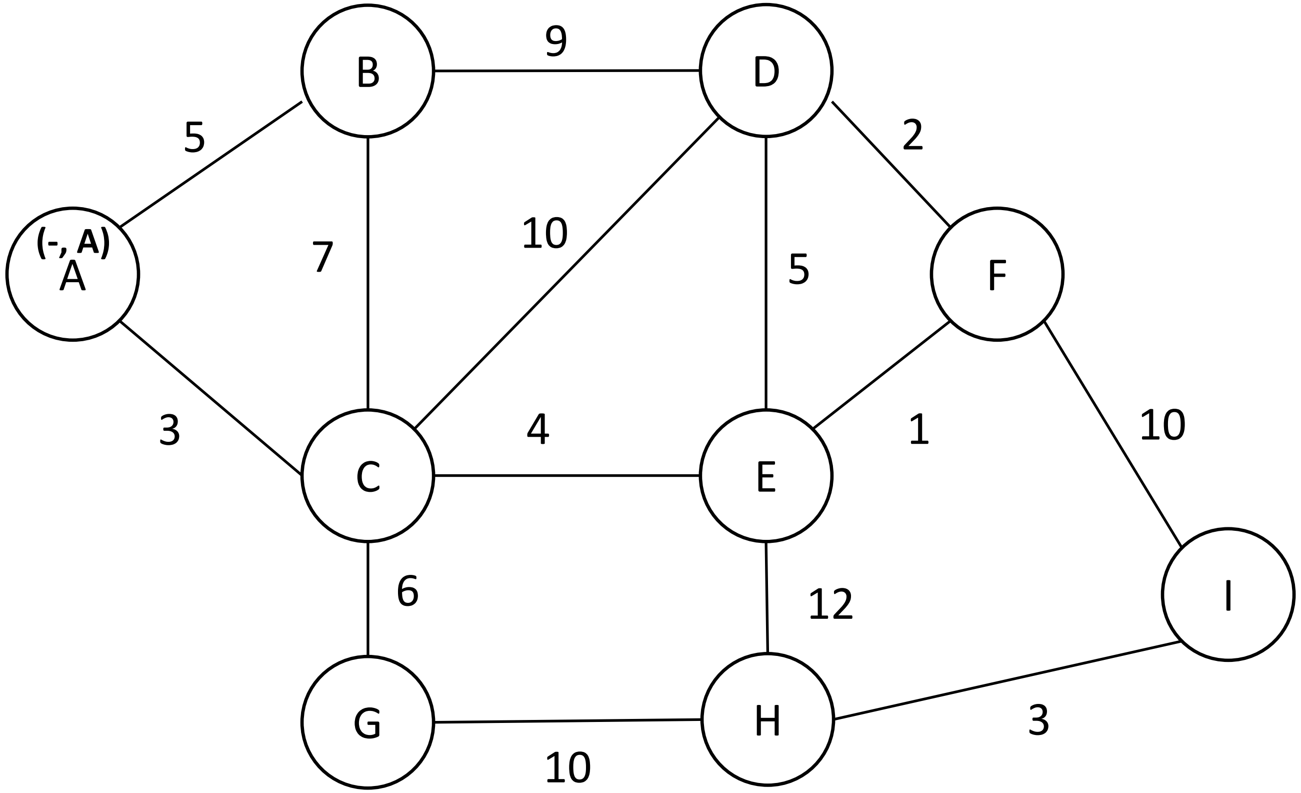
3) If vertex T has been permanently labeled stop since a shortest path from S to T has been found. If vertex T has not been labeled go to step 2.

4) When T is reached use the permanent labels on the vertices to work backwards to the origin defining the shortest path

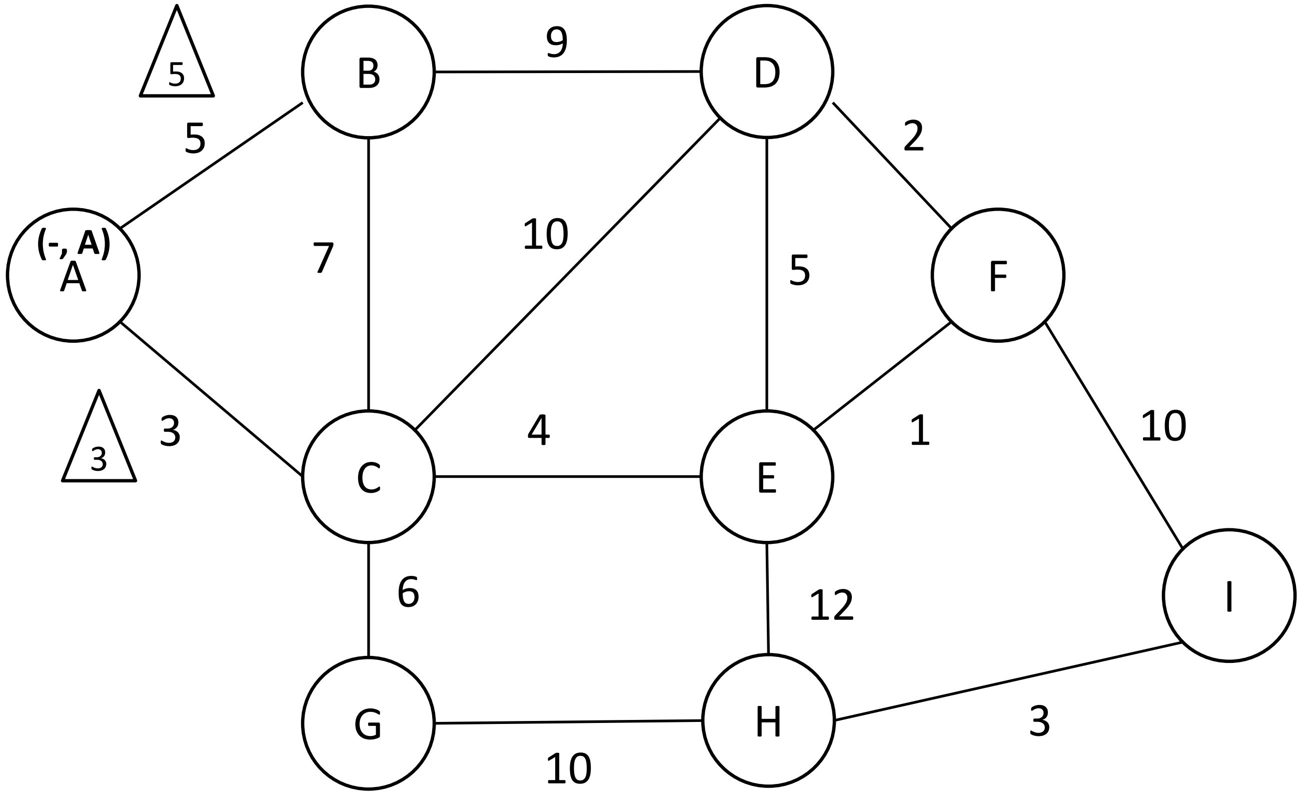
BREAK TIES ARBITRARILY. IT DOESN'T MATTER WHICH VERTEX YOU LABEL.

You might not get the same path (alternate optima) but you will get the same cost

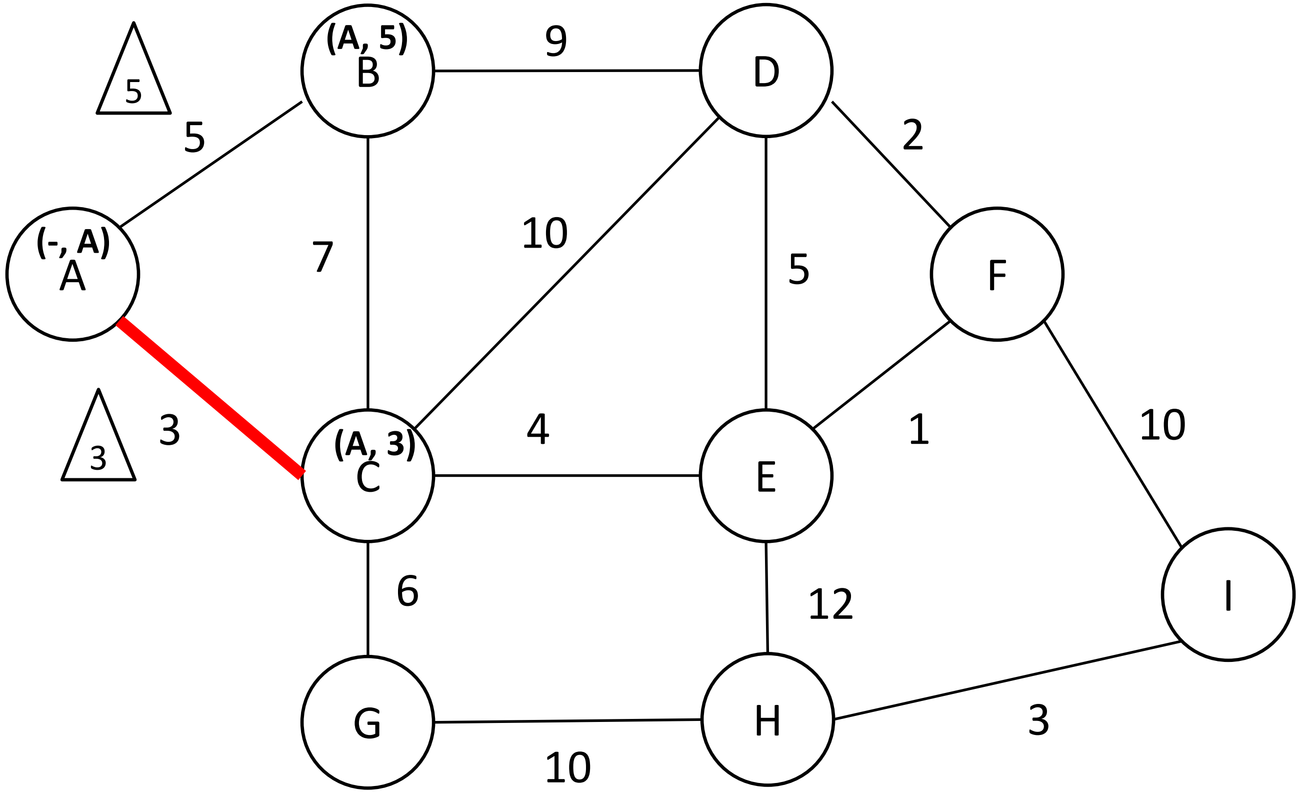
Dijkstra Shortest Path Algorithm: A to H



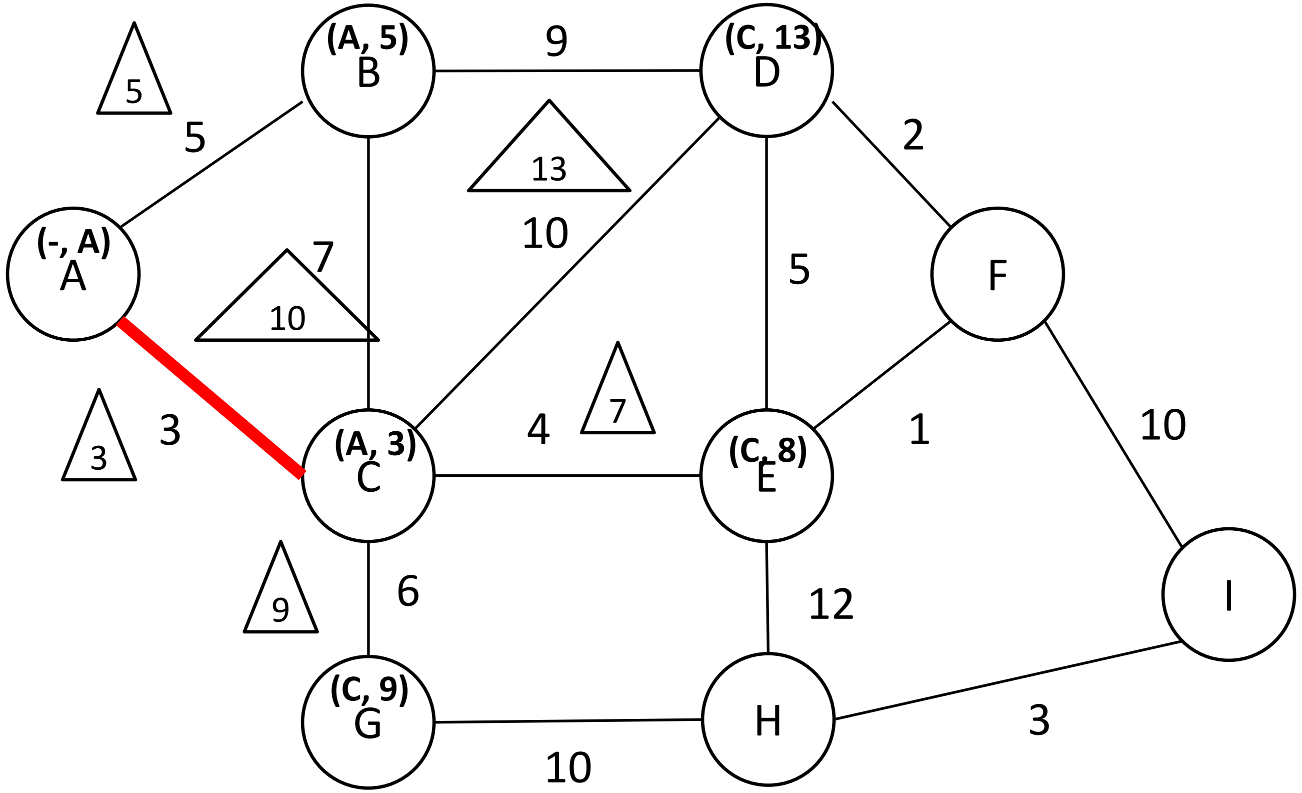
Dijkstra Shortest Path Algorithm: A to H



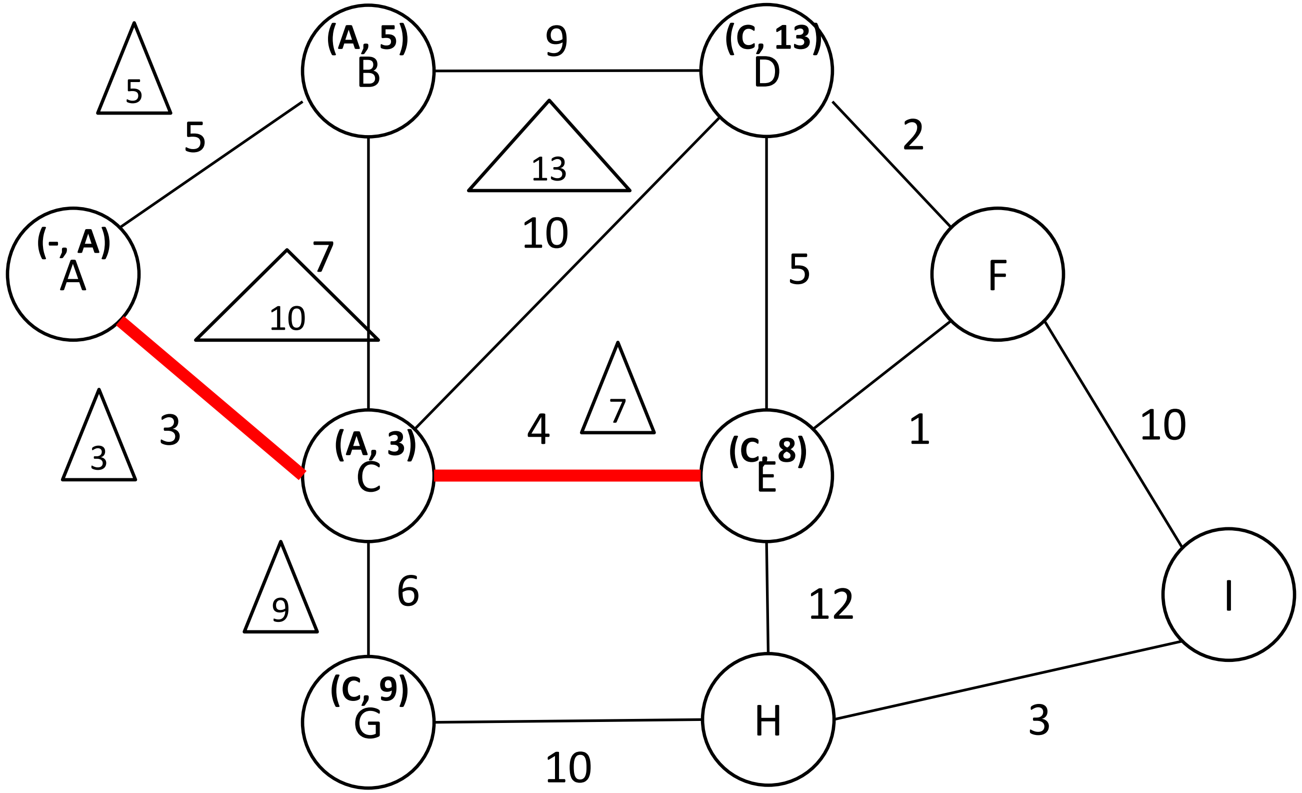
Dijkstra Shortest Path Algorithm: A to H



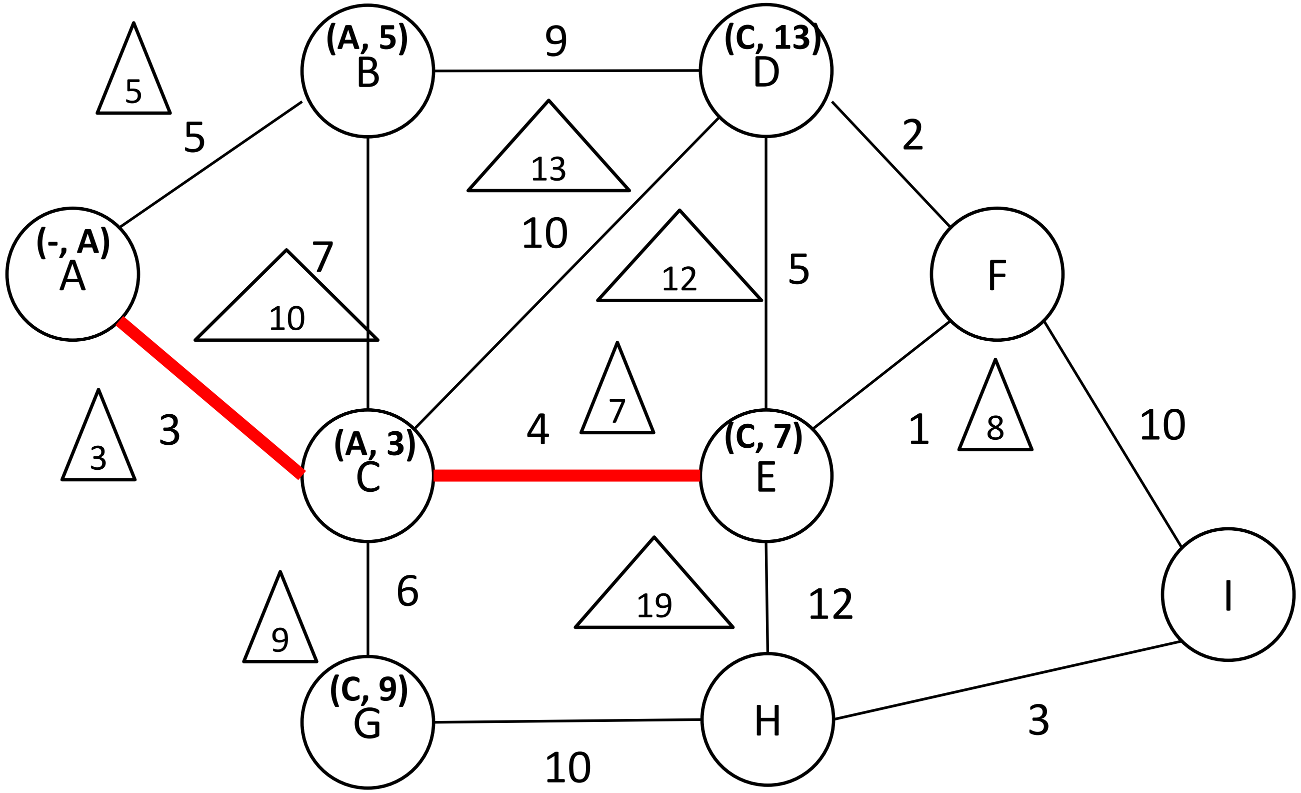
Dijkstra Shortest Path Algorithm: A to H



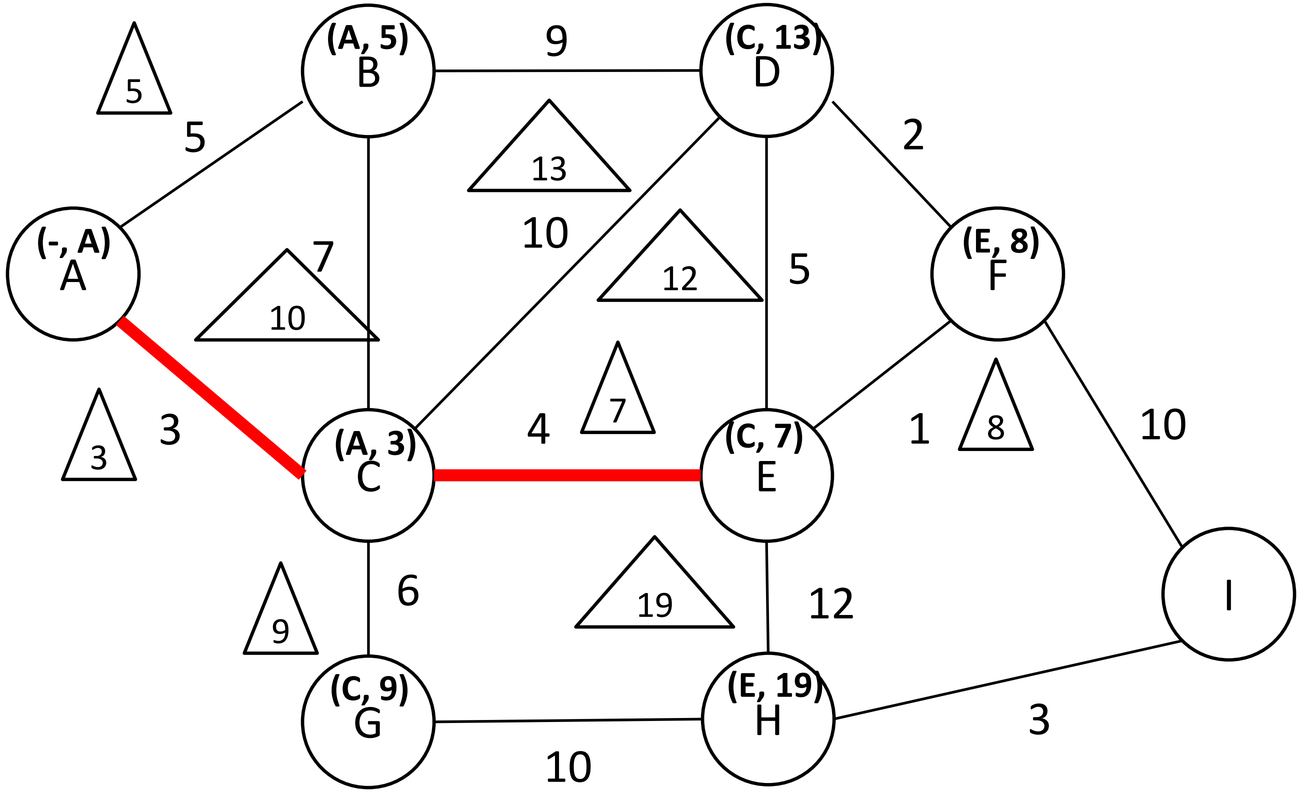
Dijkstra Shortest Path Algorithm: A to H



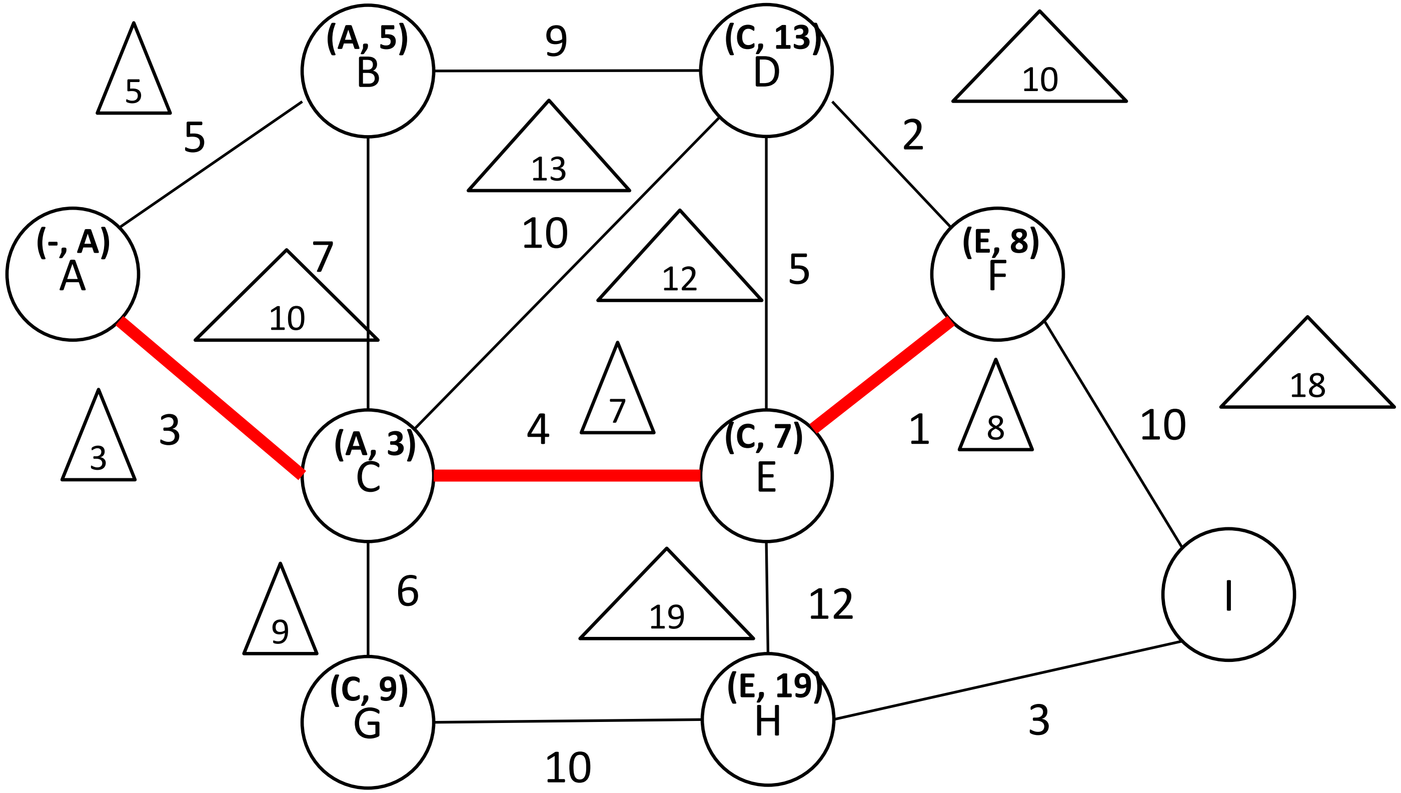
Dijkstra Shortest Path Algorithm: A to H



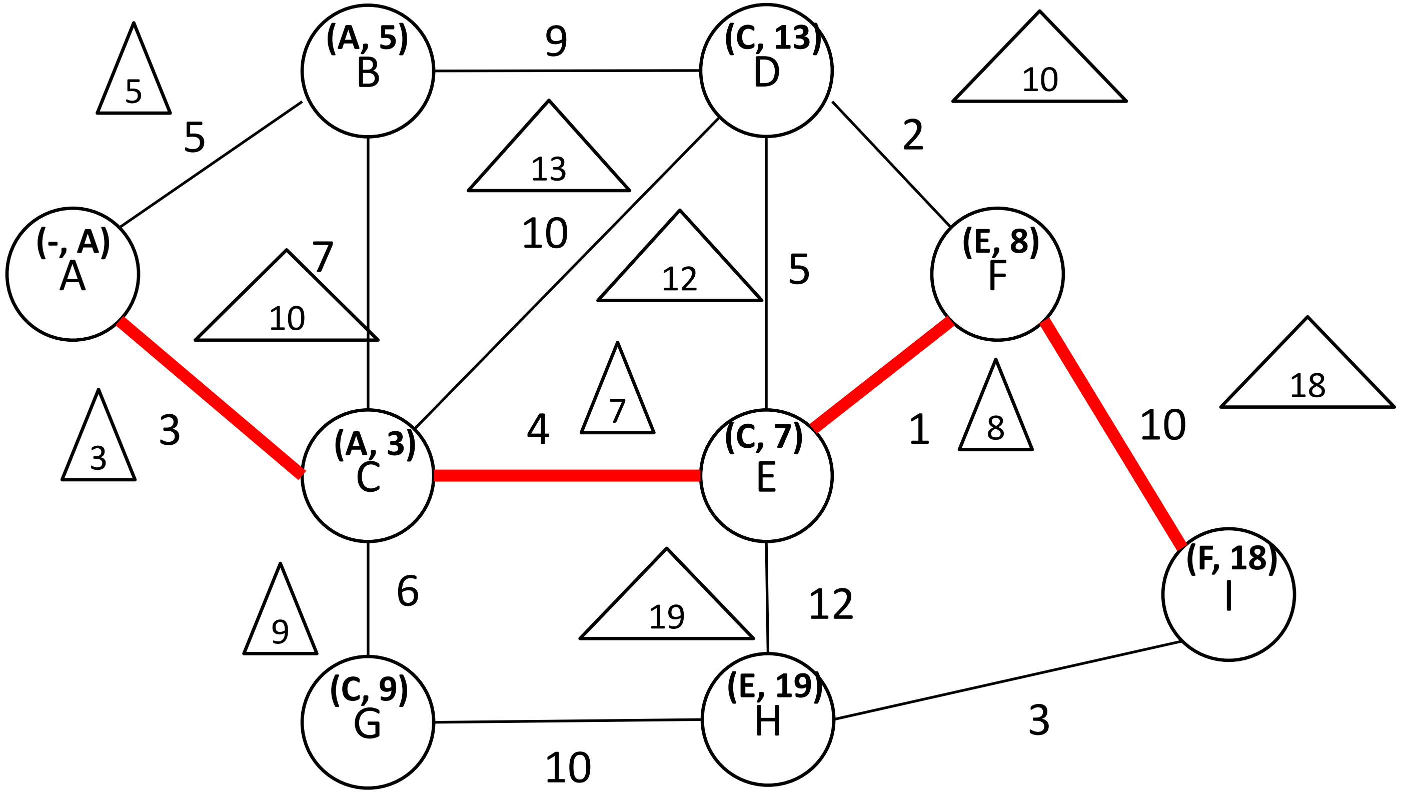
Dijkstra Shortest Path Algorithm: A to H



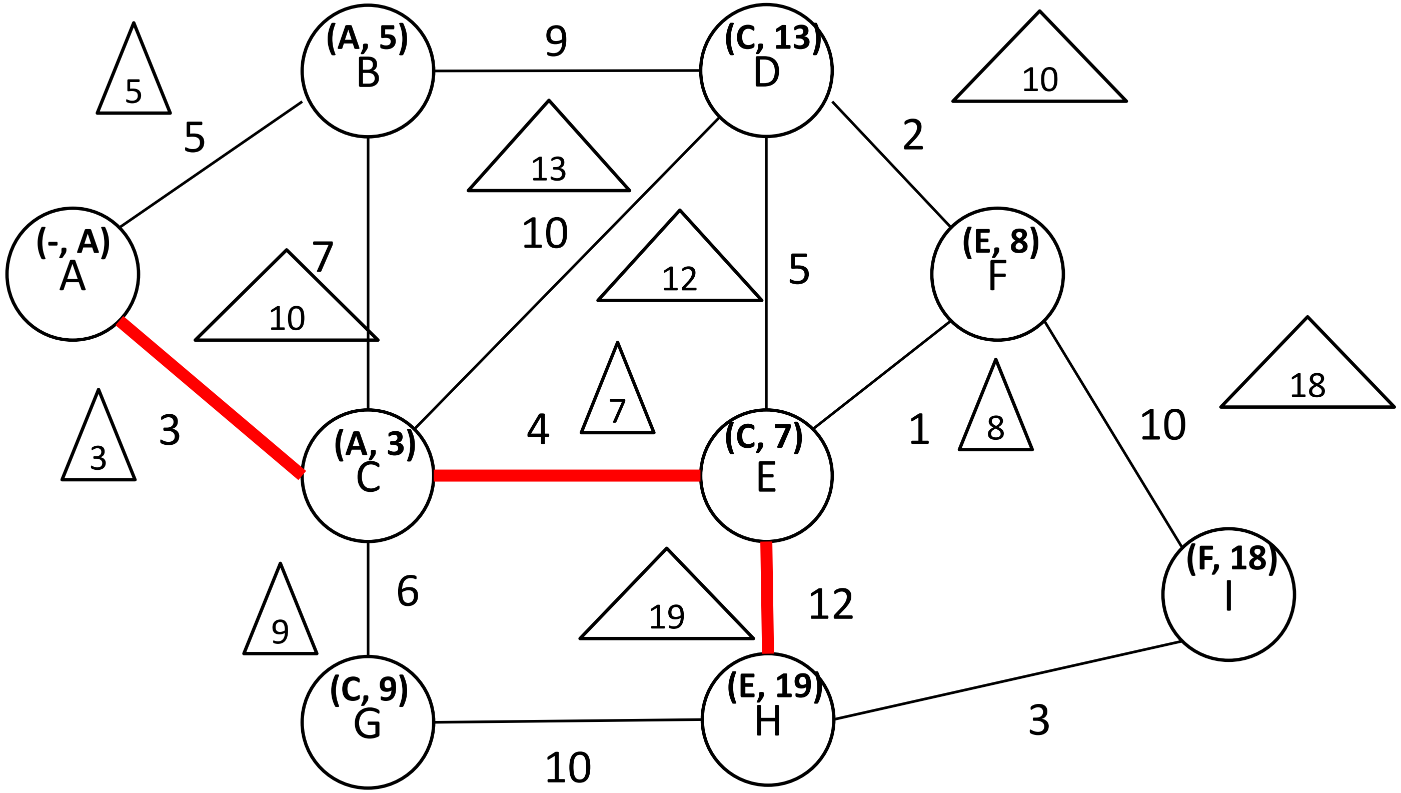
Dijkstra Shortest Path Algorithm: A to H



Dijkstra Shortest Path Algorithm: A to H



Dijkstra Shortest Path Algorithm: A to H



Dijkstra’s Algorithm (continued)

What else have we found?

- The shortest path from S to every other permanently labeled vertex
- If you want to find the shortest path from a single vertex to every other vertex in the network, continue the algorithm until all vertices have a permanent label
- Shortest paths between many permanently labeled vertices

Some variations of the shortest path algorithm

- Ford Algorithm allows negative weights
- Dijkstra’s two tree algorithm is much more computationally efficient
- All shortest path algorithms
  - Find the shortest path between every pair of nodes in the network
  - Floyd, Danzig (inventor of the simplex method)
- Just be aware that these exist