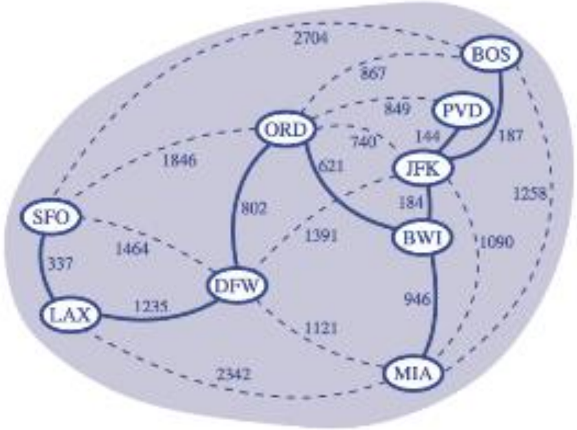
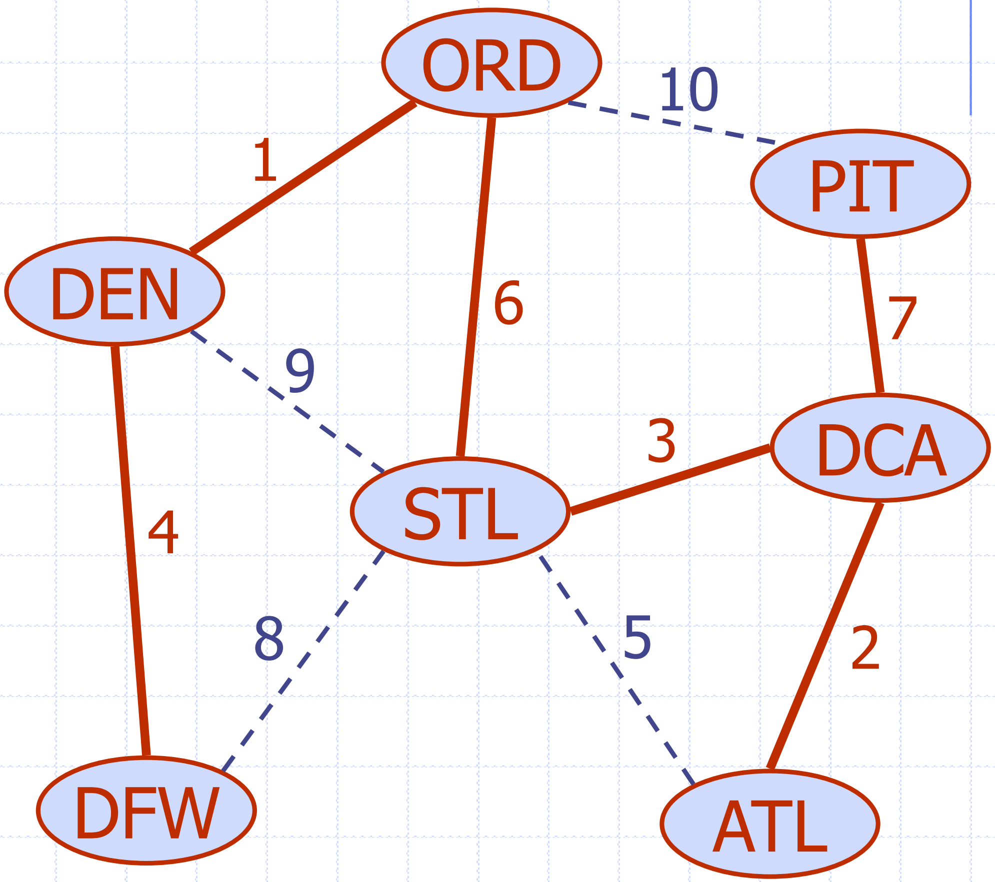
Presentation for use with the textbook Data Structures and Algorithms in Java, 6th edition, by M. T. Goodrich, R. Tamassia, and M. H. Goldwasser, Wiley, 2014. With notes from R. Solis-Oba

Minimum Spanning Trees



Minimum Spanning Trees

- Spanning subgraph
  - Subgraph of a graph ***G*** containing all the vertices of ***G***
- Spanning tree
  - Spanning subgraph that is itself a tree
- Minimum spanning tree (MST)
  - Spanning tree of a weighted graph with minimum total edge weight
- Applications
  - Communications networks
  - Transportation networks



Prim's Algorithm

```
Algorithm Prim (G,s)
In: weighted connected graph G and vertex s
Out: {compute a minimum spanning tree}
for each vertex u of G do {
    u.d ← ∞            // distance from vertex s to vertex u
    u.p ← null         // predecessor or parent of vertex u in a shortest paths tree
    u.marked ← false
}
s.d ← 0                // Distance from s to itself is 0
for i ← 0 to n-1 do {
    min ← ∞            // Find unmarked vertex u with minimum distance to s
    for each vertex v of G do
        if (v.marked = false) and (v.d < min) then {
            min ← v.d
            u ← v
        }
    u.marked ← true    // Relax all edges incident on vetex u
    for each edge (u,v) incident on u do
        if length(u,v) < v.d then {
            v.d ← length(u,v)
            v.p ← u
        }
}
```

c_1 c_1n

c_2 c_2n c_4

c_3 c_3n (adj. matrix) $c_3\deg(u)$ (adj. list)

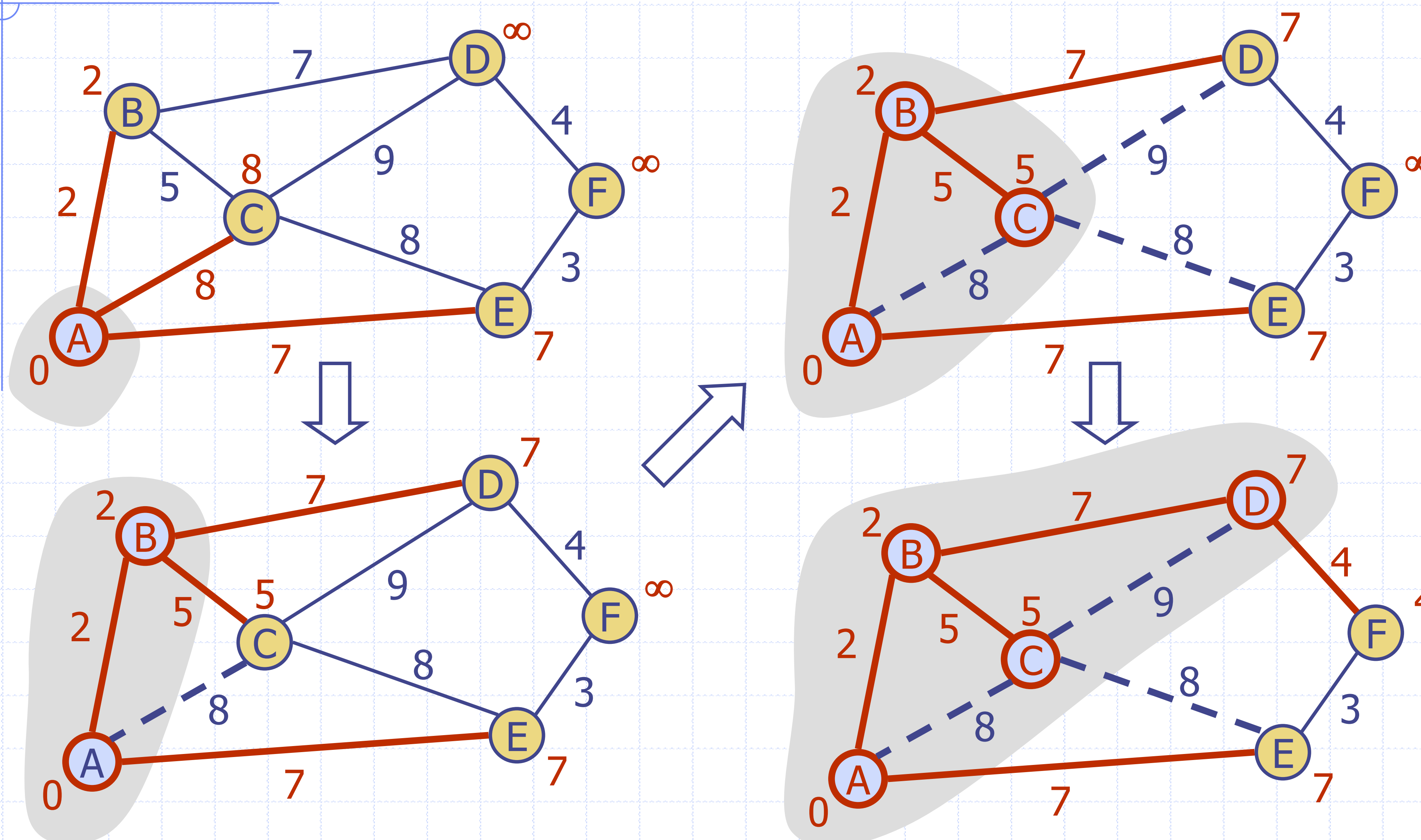
$$\sum_{u \in G}(c_4 + c_2 n + c_3 n)$$

(adj. matrix)

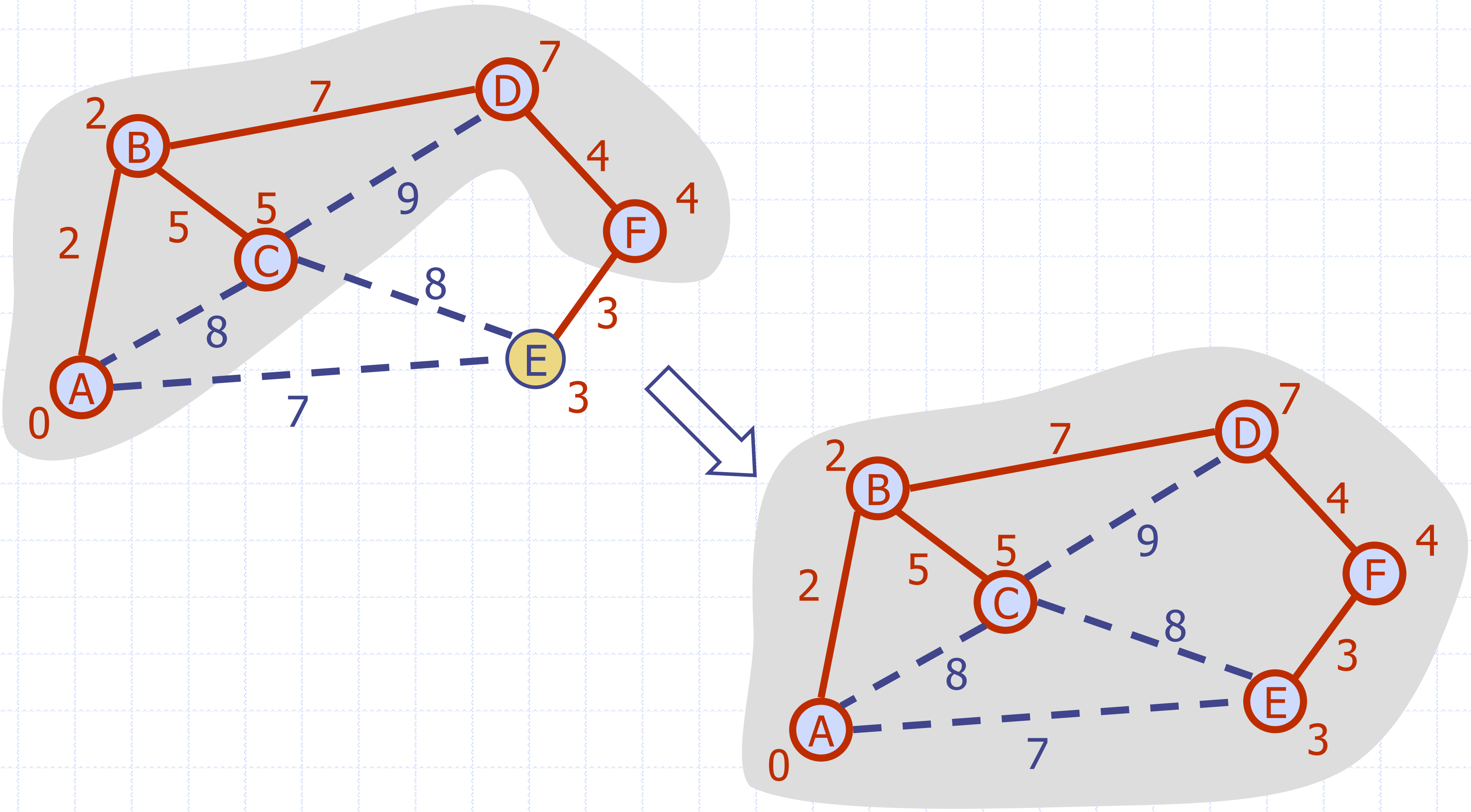
$$\sum_{u \in G}(c_4 + c_2 n + c_3 \deg(u))$$

(adj. list)

Example



Example (contd.)



Analysis of Prim's Algorithm

Using an adjacency matrix:

$$f(n,m) = c_1 n + \sum_{u \,\varepsilon\, G}(c_4 + c_2 n + c_3 n) = c_1 n + c_4 n + c_2 n^2 + c_3 n^2 \text{ is } O(n^2)$$

Using an adjacency list:

$$f(n,m) = c_1 n + \sum_{u \,\varepsilon\, G}(c_4 + c_2 n + c_3 \deg(u)) = c_1 n + c_4 n + c_2 n^2 + 2c_3 m \text{ is } O(n^2)$$