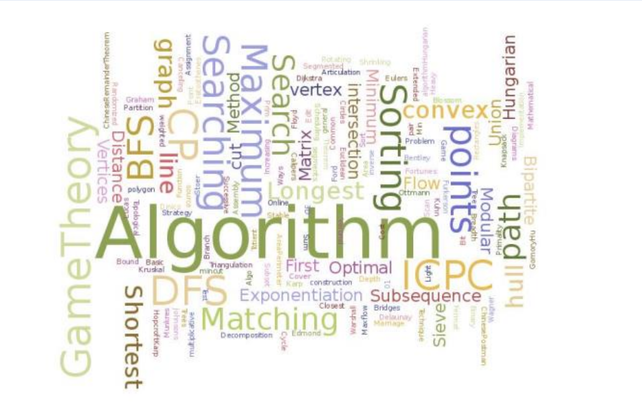
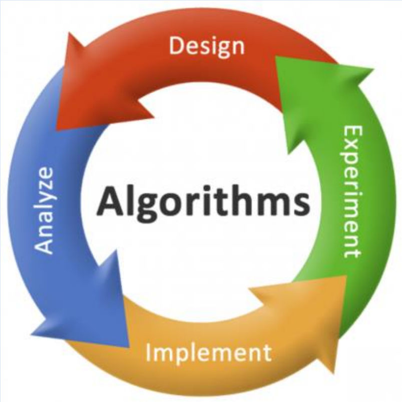
Graph Algorithms
Minimum Spanning Trees

COP 3503
Fall 2021

Department of Computer Science
University of Central Florida
Dr. Steinberg

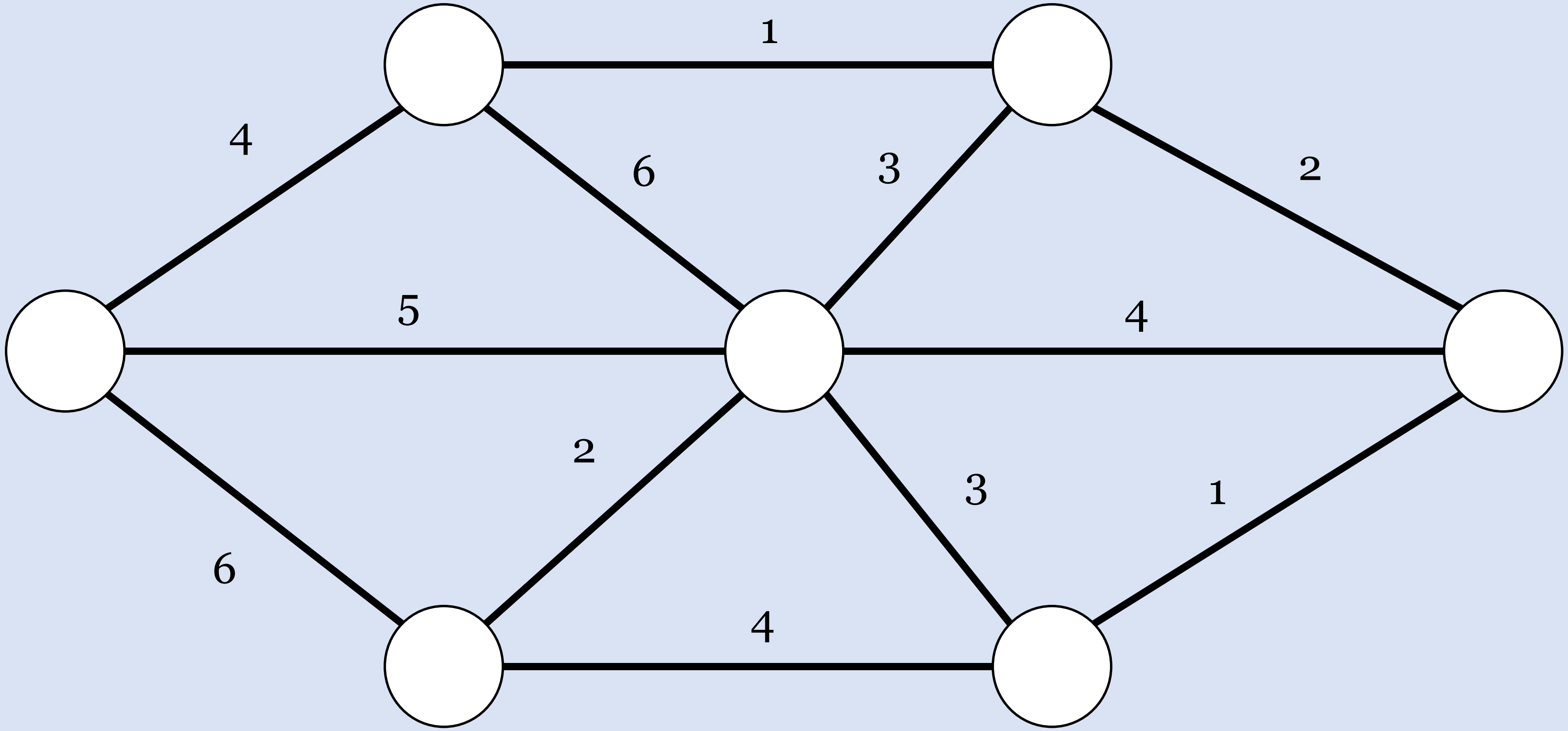




What is a minimum spanning tree (mst)?

- A set of edges from an undirected graph that contains no cycles.
  - $T \subseteq E$ such that
    - Spanning tree – meaning all edges are connected in *T* from *V*
    - $w(T) = \sum_{(u,v)\in T} w(u,v)$ is minimum
- They are primarily used in network applications.
- Input of generating a mst is an undirected weighted, connected graph *G(V,E)*
- Output is a MST

Example



Generic MST Algorithm

<u>GenericMST(G,w)</u>

A = empty set

while A is not a spanning tree

    find (u,v) a safe edge for A

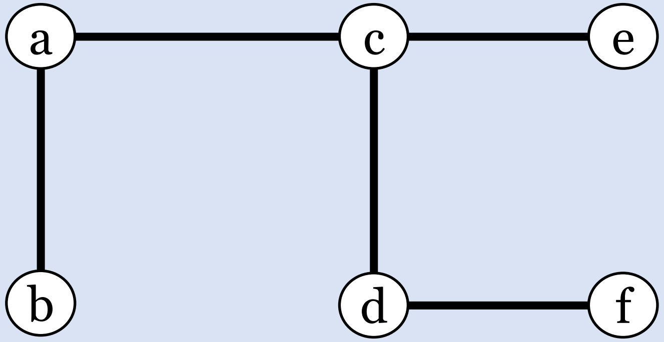
    A = A U (u,v)

return A

A safe edge for A exists if A U (u,v) is a subset of the MST

Some definitions you should know

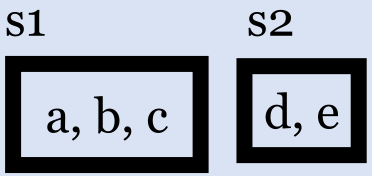
- A cut (S, V-S) is a V-S partition
  - Example
    - S = {a,b,c,d,e,f}
    - S – V = {c,d,f}
- Crossing edge is an edge (u,v) that crosses the cut if one of the end points is part of the S and S-V

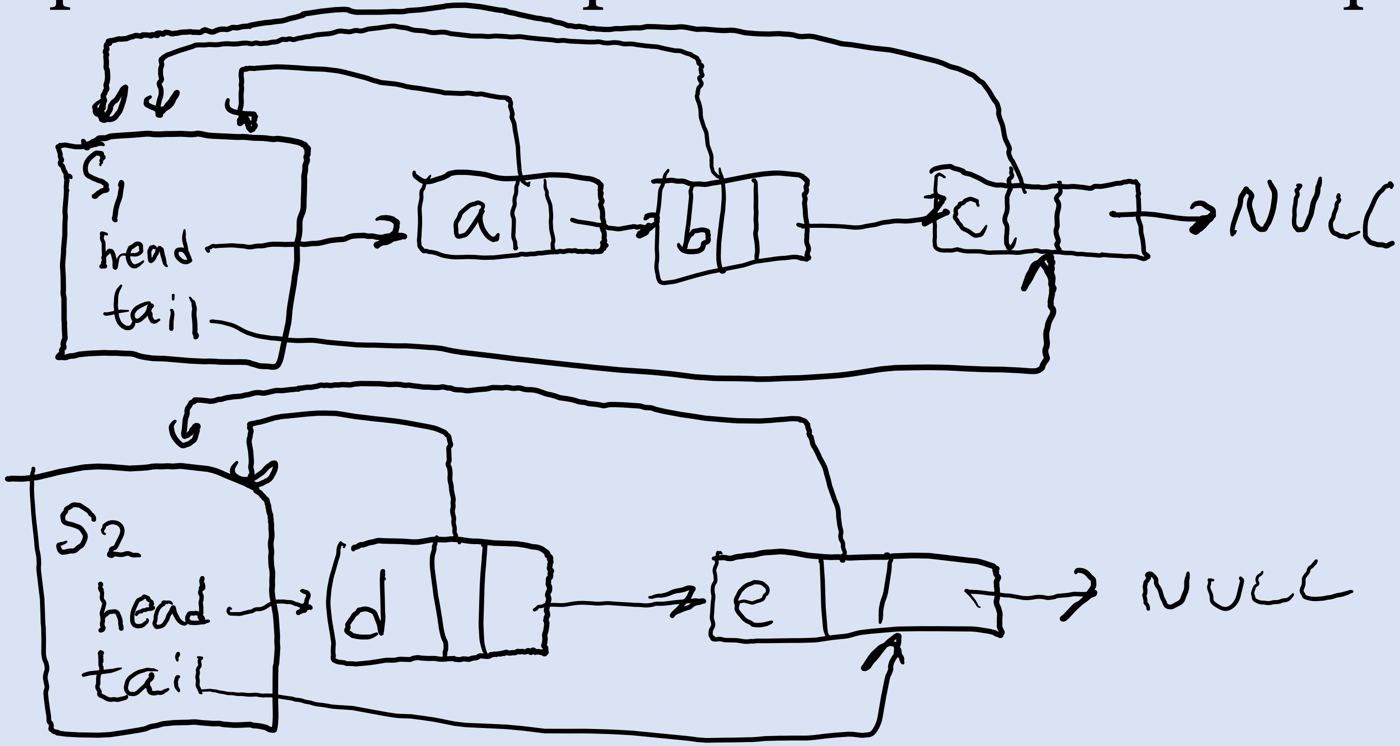


- A cut respects the set A if no edge in A crosses the cut
- A light edge is a crossing edge of minimum weight
- Theorem
  - Let A be a subset of the MST
  - Let the cut (S, V-S) be a cut that respects A.
  - Let edge (u,v) be a light edge for that cut (S, V-S)
  - Then edge (u,v) is a safe edge

Disjoint Set Data Structure

- Make-Set(x) creates a new set whose only member is x,
- Union(x,y) unites the dynamic sets that contain x and y.
- Find-Set(x) returns a pointer to the representative of the unique set containing x.





An Interesting Theorem About Disjoint-Sets

- A sequence of m Make-Set, Union, and Find-Set operations from which n are Make-Set operations take $O(m + n\lg n)$
- This will be useful for our RT analysis of the MST algorithms we will observe.

Kruskal Algorithm

Kruskal(V,E,w)

A = Empty Set

for each vertex $v \in V$

    Make-Set(v)

Sort the edges E based on weight in increasing order

for each edge (u,v) taken from sorted order

    if Find-Set(u) ≠ Find-Set(v)

        Union(u,v)

        A = A $\cup \{(u, v)\}$

return A

Example

Prim's Algorithm

<u>Prim(G,w,r)</u>
for each $u \in G.V$
    u.key = ∞
    u.π = NIL
r.key = 0
Q = G.V
while Q≠Empty Set
    u = Extract-Min(Q)
    for each v ∈ $G.Adj[u]$
        if $v \in Q$ and $w(u,v) < v.key$
        v.π = u
        $v.key = w(u,v)$

RT = $O(ElgV)$

Example