Prim's Algorithm

Q = binary heap, G = adj list $\} \rightarrow O(m \log n)$

Prim (G, c)

choose s arbitrarily

$Q \leftarrow \emptyset$, $T \leftarrow \emptyset$

for each vertex v

  $d'[v] \leftarrow \infty$

  $\pi[v] \leftarrow$ NIL

$d'[s] = 0$

for each v

  enqueue$(Q, v, d'[v])$

while $Q \neq \emptyset$

  v = extract-min(Q)

  $T \leftarrow T \cup \{(v, \pi[v])\}$

  $S \leftarrow S \cup \{v\}$

  for $(v,w) \in E$ where $w \in Q$

    if $c(v,w) < d'(w)$

      $d'[w] = c(v,w)$

      $\pi[w] = v$

      decrease_priority$(Q, w, d'[w])$

return T

INVARIANT

a) $|S|$ = # times through loop

b) $Q \cup S = V$ and $Q \cap S = \emptyset$

c) $|S| \geq 1 \rightarrow S = \{v \mid s \xrightarrow{T} v\}$

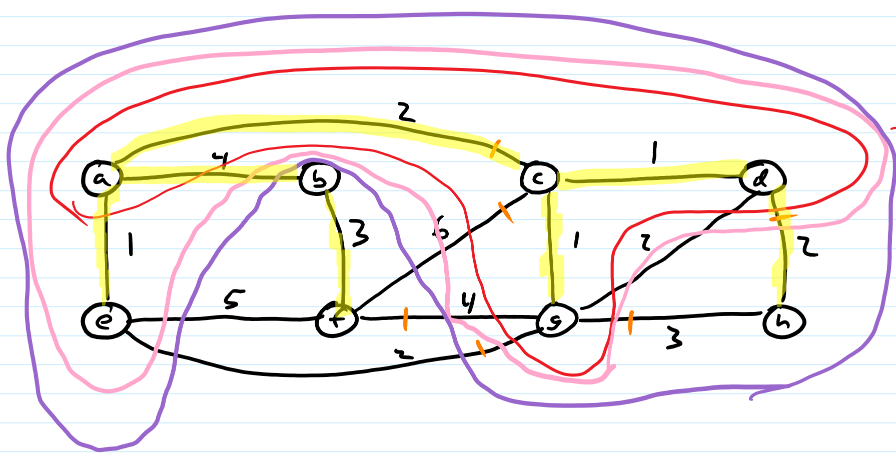
d) for $v \in Q$ i) $d'[v] = \min_{\substack{u \in S \\ (u,v) \in E}} c(u,v)$

ii) $\pi[v] = \operatorname{argmin}_{\substack{u \in S \\ (u,v) \in E}} c(u,v)$ (break ties arbitrarily)

iii) $d'[v]$ is priority of v in Q

e) T is a proto-MST

↳ of v in Q

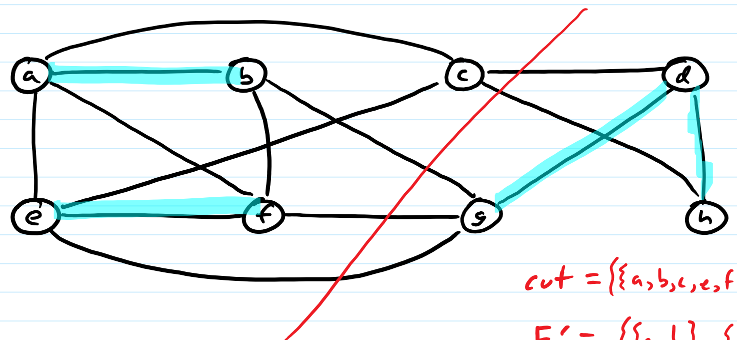


	priority
b	4
~~c~~	~~1~~
f	4
h	2

For an undirected graph $G=(V,E)$

a cut is a partition of V into S, $V-S$ where $S \neq \emptyset$ and $S = V$

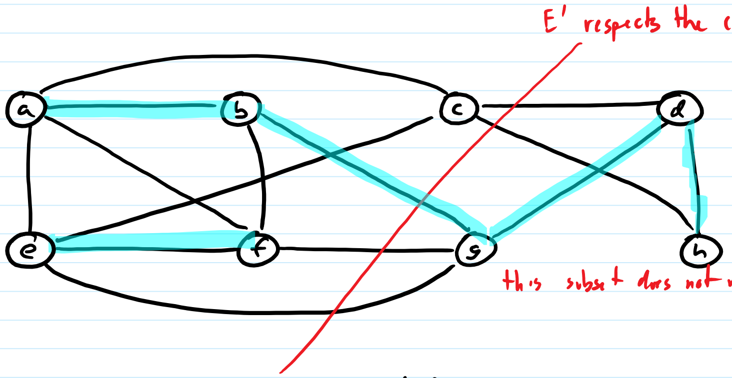
a subset of edges E' respects cut (V_1, V_2) for all $(u,v) \in E'$ either $u,v \in V_1$ or $u,v \in V_2$



cut $= \{\{a,b,c,e,f\}, \{d,g,h\}\}$

$E' = \{\{a,b\}, \{e,f\}, \{d,g\}, \{d,h\}\}$)

E' respects the cut



this subset does not respect the cut

Thm: If (Light Edge Thm)

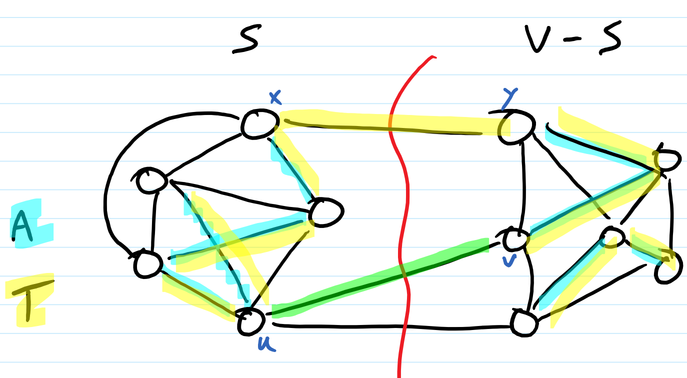
1) $G=(V,E)$ is an undirected, connected, weighted graph with $c(u,v)>0$ for all $(u,v) \in E$
2) $(S, V-S)$ is a cut of G
3) $A \subseteq E$ is an acyclic subset of E that respects $(S, V-S)$
4) A is a proto-MST (A is a subset of some MST)
5) (u,v) is min-weight edge that crosses $(V, V-S)$

then $A \cup \{(u,v)\}$ is a proto-MST

Generic-MST:

1) $T \leftarrow \emptyset$
2) find some cut $S, V-S$ that T respects
3) find min-weight edge across $(S, V-S)$
4) $T \leftarrow T \cup \{(u,v)\}$

Inv: T is a proto-MST



proto-MST

path-MST

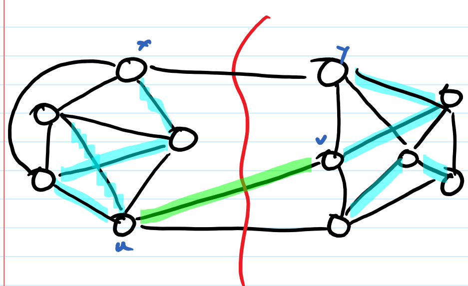
Proof: Suppose $G, S, A, (u,v)$ satisfy 1-5 above.

Find MST T so that $A \subseteq T$

Two cases: 1) $(u,v) \in T$. Then $\underbrace{A}_{\subseteq T} \cup \underbrace{\{\{u,v\}\}}_{\subseteq T} \subseteq T$

2) $(u,v) \notin T$

Then there is some edge (x,y) on path $u \rightsquigarrow v$ in T so that $x \in S, y \notin S$



Let $T' = \underline{T - \{\{x,y\}\}} \cup \{\{u,v\}\}$

↳ has 2 connected components: one with u, x; one with y, v

(u,v) joins those connected components

$u \rightsquigarrow x \not\rightsquigarrow y \rightsquigarrow v$

So T' connects G (if $v_1 \overset{T}{\rightsquigarrow} v_2$ doesn't use (x,y) then same path connects v_1, v_2 in T'; otherwise do $v_1 \rightsquigarrow \underbrace{x \rightsquigarrow u \rightarrow v \rightsquigarrow y}_{\text{replace } x \to y \text{ with path in } T'} \rightsquigarrow v_2$ to connect v_1, v_2 in T')

And T' is acyclic (T is acyclic; any cycle in T' can be transformed into a cycle in T by replacing $u \to v$ by path $u \rightsquigarrow v$ in T)

And T' is spanning tree (acyclic and connects)

$x \le \_\_ \le \_\_ \le \_\_ \le x$

$x = \_\_ = \_\_ = \_\_ = x$

Furthermore, $c(T) \le c(T')$ (T is MST, T' is a ST)

$= c(T) - c(x,y) + c(u,v)$

$\le c(T)$ $\quad (c(u,v) \le c(x,y))$

so all $\le$ are $=$ and hence $c(T) = c(T')$

so T' is a MST and $A \cup \{\{u,v\}\} \subseteq T'$

A: $\subseteq T - \{\{x,y\}\}$, $\subseteq T'$; $\{\{u,v\}\}$: $\subseteq T'$

Cut Property: Given undirected, connected G with distinct positive weights, if (u,v) is min-weight edge across some cut $S, V-S$ then (u,v) is in every MST of G (and hence there is a unique MST)

Proof: use same edge exchange as for Light Edge Theorem

Kruskal's Algorithm

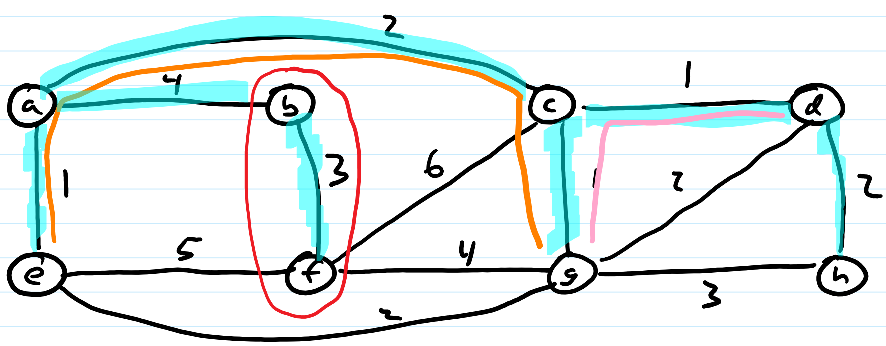
Kruskal's Algorithm : consider edges in order of ↑ weight
add edge if connects two different components of proto-MST

$T \leftarrow \emptyset$
sort edges in order of increasing weight
for each edge (u,v)
  if u,v in different connected components
    $T \leftarrow T \cup \{(u,v)\}$

$O(m \log n)$

BFS on T $O(m+n) = O(n)$ since $|T| \le n-1$
m times $O(m \cdot n)$ total

need something else



(a,e) (c,g) (c,d) (a,c) (e,g) (g,d) (d,h) (b,f) (g,h) (a,b) (f,g) (e,f), (f,c)

connected components $\{a\}$ $\{b\}$ $\{c\}$ $\{d\}$ $\{e\}$ $\{f\}$ $\{g\}$ $\{h\}$

Init : a) b) c)

Maintenance: Suppose a, b, c T before loop

a)

b)

c)

Termination: At termination, all edges in G have been examined.

For any pair of vertices $(u,v) \in E$, (u,v) are connected in G

there is a path $u = x_1, \dots, x_k = v$ in G

for all i, x_i, x_{i+1} are connected in A

(u,v) are connected in A

$\therefore$ A spans G

A is a proto-MST

A is an MST