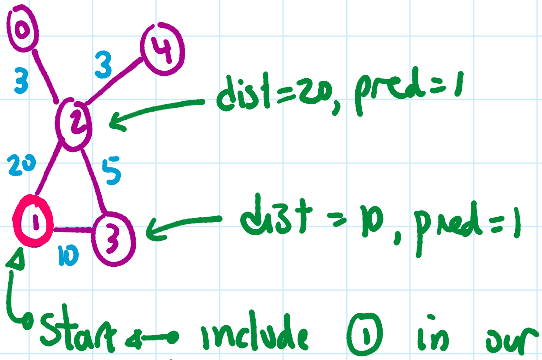
CMSC351-2023-11-27

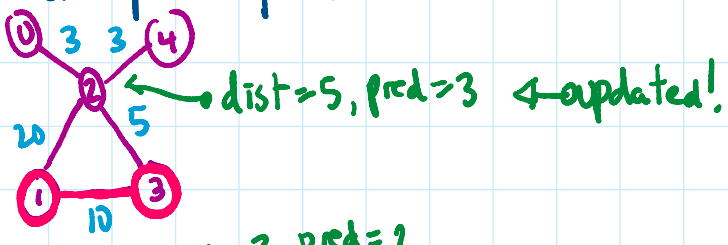
Monday, November 27, 2023 8:07 AM

Prim Continued (to reduce time complexity!)

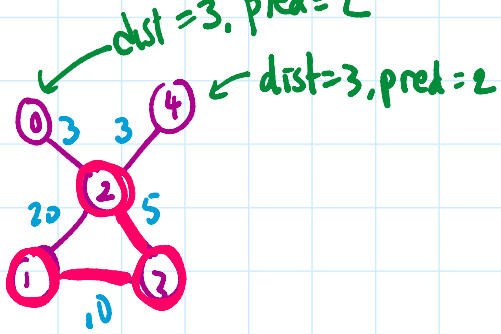
(D) We start w/ our vertex s.
For each vertex y adj to s, assign dist[y] = edge weight.
pred[y] = s



Then: Choose the vertex not in T w/ min dist, call that x
include that, and the edge connecting it to its pred, in T
Then, for all vert adj to x which are not in T,
update dist if the dist to x is shorter. If so,
also update pred.

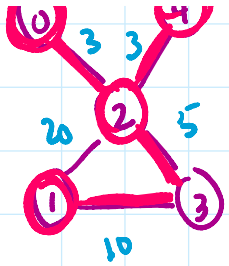


Repeat!



(note: dist is storing distances from the tree to the vertices which are adj. to vert. in the tree.)

Repeat 2x!



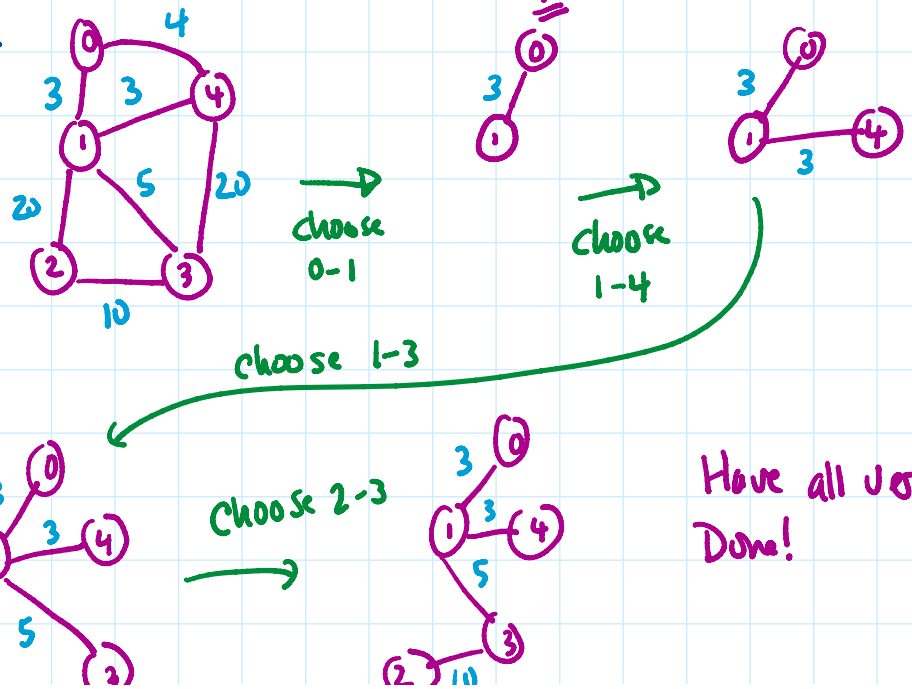
done!

Time Complexity: To find the vert. w/ min dist which is not in T is just $\theta(V)$. Then to update adj. vert. that's $\theta(V)$ as well, but those are in series.
So that's $\theta(V+V) = \theta(V)$.
But then that's done V times (to get all vert.) for a total of $\theta(V^2)$.

Kruskal's Algorithm!

① Idea: Given a graph, we proceed as follows:
Until we have all vertices, do the following. (and enough edges)
- pick an edge of min. weight whose inclusion in T does not form a cycle.
include it.

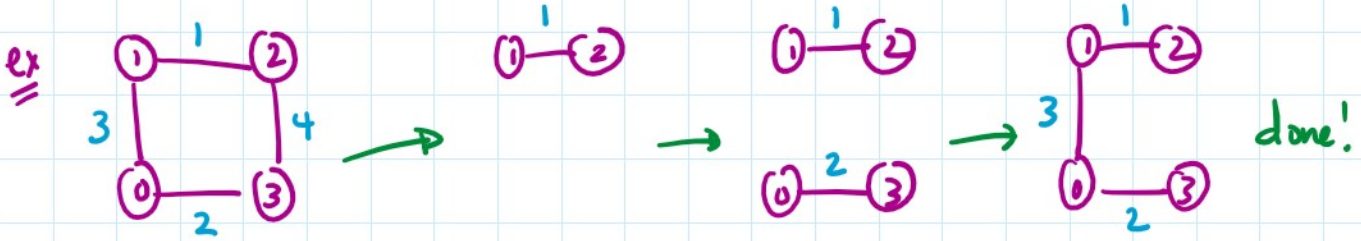
② Example



③ Note

(A) This is a greedy algorithm.
Not all greedy algs yield an optimal result (coin change!)
but this one does! See proof in Latex Notes!

(B) The tree is not nec. even a connected graph
until the end!



③ Time Complexity:

Complication: given a graph (our T), how can we tell,
if a new edge were to be included,
whether this forms a cycle?!

of course there are ways, which we won't go into detail with
except to say:

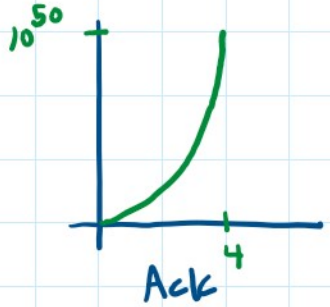
(A) Using some basic data structures such as min heaps, etc.
Kruskal's Alg can be done in $O(E \lg E)$.

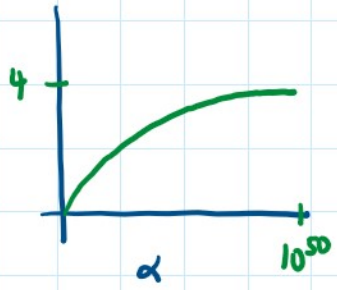
(B) In addition since $E \leq C(V,2)$ (in any simple, conn, undirected graph)
we have $E \leq V^2$ so we can also get $O(E \lg V^2)$
but $E \lg V^2 = 2E \lg V$ so it's also $O(E \lg V)$.

(C) The best way to do cycle detection is w/ a data
structure called a Disjoint Set Data Structure.
We can bring this to $O(E\,\alpha(V))$.
Note: α is the inverse of the Ackermann Function

We can bring this to $O(E\,\alpha(V))$.

Note: α is the inverse of the Ackerman Function.





So basically, unless V is stupidly large, $\alpha(V) \leq 4$ and so it's "basically constant".

So $\approx O(E)$.