Subject: Prims
Posted by nltollman on Wed, 23 Nov 2016 01:54:59 GMT
View Forum Message <> Reply to Message

I am currently typing out Prims and am confused on what is modified/returned at the very end. Is it the graph G that is being passed in that's overall being modified? Or are we creating a tree A and inserting vertices as we go along Prims then returning that at the end?

If G is the case, then is G a combination of the AdjMatrix and the array of vertex objects?

If I choose to not passing in a G, but rather my AdjMatrix and array separately, what should be modified/returned at the end of Prims?

If A is the case, then I figure that the first vertex object with infinity as the key (r in Prims pseudocode) is set to be A's root. Then, from there on out, we add u into A and its children, and do a comparator between what's in A to do a subheap union, if necessary (because from what I am looking at, this is the ONLY binomial heap function that is not being called at all from anywhere else, since we are no longer using the function deleteFromBinHeap)?

Subject: Re: Prims
Posted by lusth on Mon, 28 Nov 2016 15:47:05 GMT
View Forum Message <> Reply to Message

I just store the predecessor in each vertex object that comes out of the priority queue. My Prims algorithm returns the starting vertex and then I print out the spanning tree rooted at the starting vertex using the predecessor fields.