question 72 views

Optimal Solution for Q4 on exam 2

I lost most of my marks in Q4 , which was supposedly the easiest question on the exam.
Could somebody share the optimal logic here?

exam hw4

Updated 1 day ago by Ravindra Kumar Yadav

the students' answer, *where students collectively construct a single answer*

Dijkstra from z^* in G to discover the distances $d_G(z^*, v)\ \ \forall v$, then Dijkstra from z^* in G^R to discover the distances $d_{G^R}(z^*, v) = d_G(v, z^*)\ \ \forall v$, then compare. You may have misread the question thinking it was asking for all pairs?

Updated 22 hours ago by MarkBenjamin1@gatech.edu

the instructors' answer, *where instructors collectively construct a single answer*

Just to point out, isn't the solution almost identical to the homework problem for shortest paths through v_0?
I was worried it's too similar...

Updated 4 hours ago by Eric Vigoda

followup discussions *for lingering questions and comments*

Resolved Unresolved

Ravindra Kumar Yadav 22 hours ago
My solution was this:
for all vertices v calculate dist(v,z*) and dist(z*,v), then apply the comparison. Isn't it similar to what the optimal solution says.

MarkBenjamin1@gatech.edu 21 hours ago how do you calculate $dist(v, z^*)\ \ \forall v$?

for that matter, how did you say you calculate $dist(z^*, v)\ \ \forall v$? From your description, you may have omitted to specify an algorithm for that

Resolved Unresolved

Scott Allen Quinn 14 hours ago
The problem I had was, always used Dijkstra's as shortest path from a single vertex u to a single vertex v. However, it actually returns shortest path from u to all vertices in the graph. As a result, only need to run it once on the graph G and once on the reverse graph G^R.

Then just compare the distances between the two runs.

Anonymous 11 hours ago Hey if there's a path from v to z* and z* to v doesn't it mean there's a cycle?

Santiago L. Valdarrama 11 hours ago Remember you are looking for a path from z* to v **in the reverse graph**. This is key.

Anonymous 10 hours ago thats exactly what i did at first and then i crossed it out because it didn't seem to make sense.....im trying to figure out why we need to reverse the graph

we're given a directed graph...find all v for which distance v to z* and z* to v satisfy some criteria.....doesnt that mean there has to be a path from v to z* AND z* to v?

what am i missing?

Brian Xia 10 hours ago +1 this single misconception costs 15 points :((

Santiago L. Valdarrama 10 hours ago You need to find all vertices where $dist(v, z*) > 2dist(z*, v)$

You can easily find the distance from $z*$ to all the other v vertices by running Dijkstra one time.

But how do you find the distance from every v to $z*$?

1. First solution: For every v, run Dijkstra and get the distance from v to $z*$.
2. Second solution: Reverse the graph, and run Dijkstra one time to get the distance from $z*$ to every v in the reverse graph. These distances will be equivalent to the distance for every v to $z*$ in the original graph.

Both solutions work, but the second solution needs only one Dijkstra while the first solution needs $|V|$ Dijkstras.

Does that make sense?

Anonymous 10 hours ago santiago based on what you just posted, i wrote down solution 2., crossed it out and wrote down the less efficient solution 1 instead... but lost 15 points....unless im missing something else?

i said run dijkstra from z * to v and v to z* and output vertices which satisfy the criteria.

is that what you are saying for your solution 1?

Santiago L. Valdarrama 9 hours ago If you said just like that, then it is incorrect. You had to specify that you needed to run Dijkstra for every v, and then take the min distance from v to z*. The way you said it implies that only one run of Dijkstra is sufficient.

Anonymous 9 hours ago No I didn't state it exactly like that... just an abbreviated version...but even if I did I believe v represents a set of vertices according to the problem statement so that would be correct... anyway I'll just ask the grader....

Nolan Capehart 8 hours ago I think the key point is that you need to run Dijkstra on the reverse graph so that you can find dist(v,z*) for ALL v simultaneously. That's Santiago's #2.

So, there are several possible mistakes:

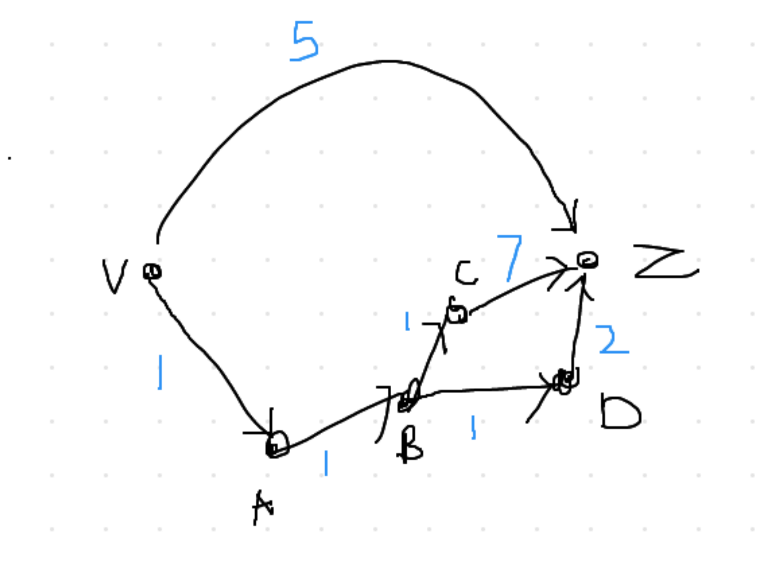
1) Don't mention Dijkstra at all.

2) Mention Dijkstra for finding dist(z*,v), but don't mention Dijkstra for dist(v,z*).

3) Mention Dijkstra for finding dist(z*,v), and also for dist(v,z*), but leave the impression that you want to run it on the original graph to find dist(v,z*), which would imply running it many times.

◉ Resolved ○ Unresolved

Collin Lee 4 hours ago
Imagine the following directed graph:



What I was thinking was that if you did Dijkstra on the reverse graph, the vertex A when going Z -> V would have value of 4 (the shortest path value via Z->D->B->A and thus we'd miss the 2x longer path Z->C->B->A->V. This is why I was thinking we had to run Bellman Ford instead on the reverse graph.