

HW - 10

1.) we have a graph G and $ISP = j$ -th shortest path.

$ISP(\text{graph } G, k, x, y)$

list path = $\{ \}$

while path.length $> k \ \&\& \ \text{path} \{$

 run shortest path alg on G & get path btw x & y .

 add shortest path from G .

 remove shortest path from G .

 run Dijkstra shortest path alg on G and get shortest path
 btw x and y .

$\}$ return . path.

2. two bugs B_1 and B_2

$B_1 = G$ for initial to final node

$B_2 = G$ that do not have red edge immediately followed by a yellow edge.

Step 1: Make two FA M_1 and M_2 for B_1 and B_2 , respectively that accept a walk from their two int's.

Step 2: obtain the cartesian product M_1 and $M_2 = M_3$, where M_3 is a FA that accept all paths in G that start from the initial node and end at the final node and do not have a yellow edge immediately after a red edge.

Step 3: then use ~~shortest~~ shortest path algorithm on graph M_3 to obtain the path with the shortest ~~weight~~ weight.

4. Similar steps are taken as if the walk were increasing weight, but this time, the shortest path method is changed to accommodate multiplication. To make this work, we must utilize the log property and swap out the weight "a" with $\log(a)$: $\log(x * y) = \log(x) + \log(y)$. When the current node is x, meaning will use $\log(x)$, and when the current node is y, meaning will use $\log(y)$. The shortest path algorithm can be used with this modification to find the shortest path.