James Stallkamp

CS 325 hw 3

1)

A)

AE

EB

BC

CD

CG

GF

DH

B)

A: 0

E: 4

B: 5

C: 11

D: 12

F: 15

G; 15

H: 24

2)

3)

CS425

1

CS401

1

CS375

1

CS3701

CS3511

CS325

CS222

CS221

CS150

CS151

Mth201

MTH2000

A

B

* Math 200
* Math 201
* CS 150
* CS 151
* CS 221
* CS 222
* CS 325
* CS 425
* CS 351
* CS 370
* CS 375
* CS 401

C

* Term1: Mth 200 & CS 150
* Term2: Mth 201 & CS 151
* Term3: CS 221, CS351, & CS370
* Term4: CS222 & CS 325
* Term5: CS375 & CS425
* Term6: CS 401

D The longest path in the DAG is

CS150, CS151, CS221, CS222, CS375, CS401. To find the longest path of a DAG you need to sort it topologically and then compute the length of the longest path ending at each vertex.

4) This pseudo code will iterate over the vertices without color to assign them a color that is opposite of the adjacent vertices.

Main():

List=[]

While (there are uncolored vertices)

Selected uncolored

Color(n,l)

Return list

Color():

If n is colorless

For each vertex m with edge to n

If m == color1 and m == color2

Return “graph is not two colorable”

If m == color1:

n=color2

else if m ==color2

n=color1

else

n = color1

color(m)

list.append(n)

Main()

b)Each vertex and edge are checked once and all operations are constant so the algo is O(V+E)

5)

A The algorithm I would recommend is Dijkstra’s.

This algorithm would reach the vertices in the following order. Starting at G, E, H, D, B, C, F, A

B Fist Dijkstra’s algorithm would be called on every vertex to find the smallest distance . The vertex that yields the smallest maximum distance would then be the optimal station location. The run time of Dijkstra’s algorithm is O(r log(f)), this will be run f times so the overall run time will be O(f \* r \* log(f)).

C In the given graph the optimal location for the firehouse would be at intersection E where the smallest longest path distance would be E to A which is a distance of 10.