Homework - 5

(Graph Problems)

CS300 – Data Structures

FALL 2021/2022

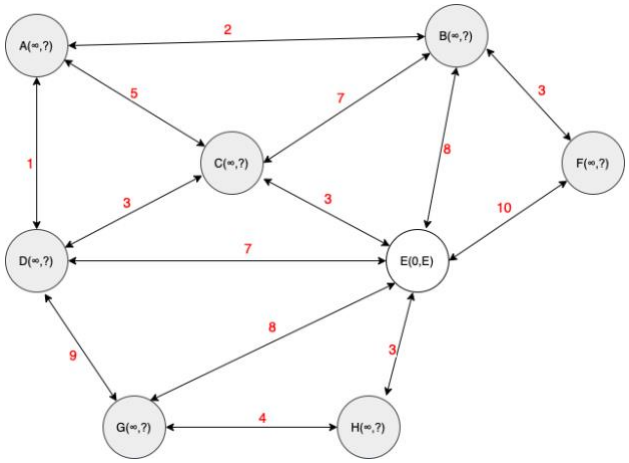
Submission Report

by

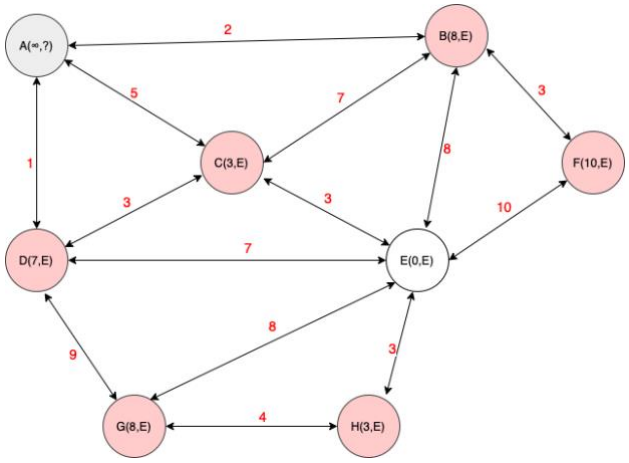
Furkan Kerim Yıldırım

28138

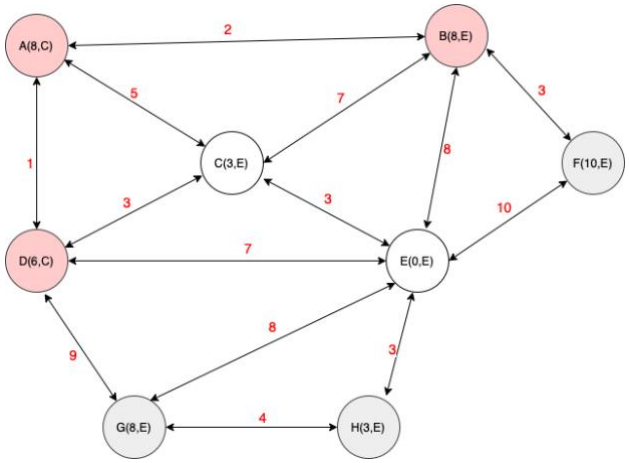
Question 1: First, the starting vertex (*E*) is selected and painted white.



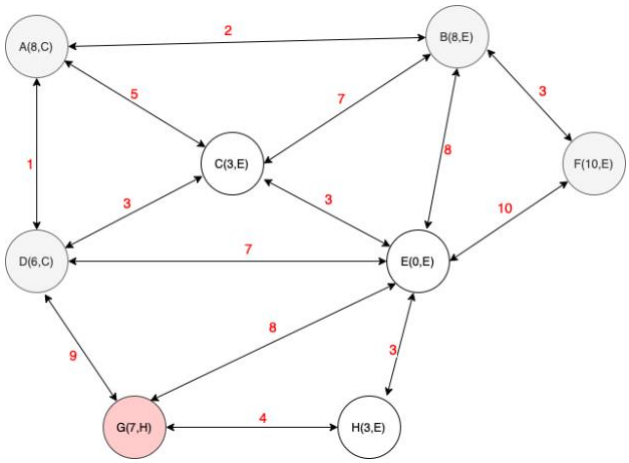
Then, the vertices adjacent to *E* are colored pink and their distance to *E* is calculated.



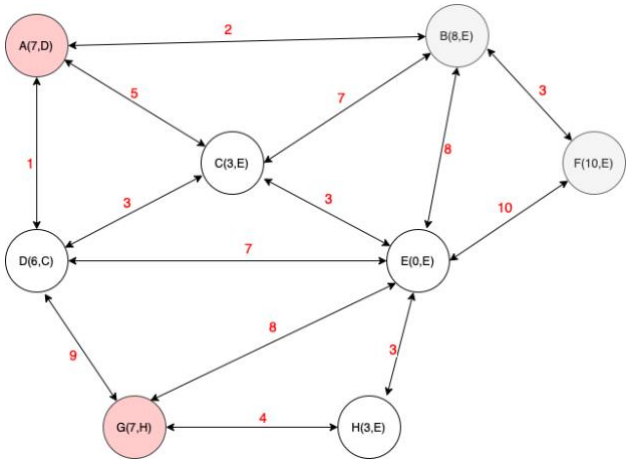
After the vertices adjacent to *E* are calculated, all the vertices are painted back to gray, the vertex with the smallest value among the vertices whose length is calculated (*C*) is painted white, and the vertices adjacent to this vertex (*C*) are painted pink and their lengths are calculated. At this stage, the length value of the *D* vertex is updated as it will be smaller when going over the *C* vertex. The length value of vertex *A* is calculated, and no update is made because it will be larger when going over vertex *C* on other neighboring vertices.



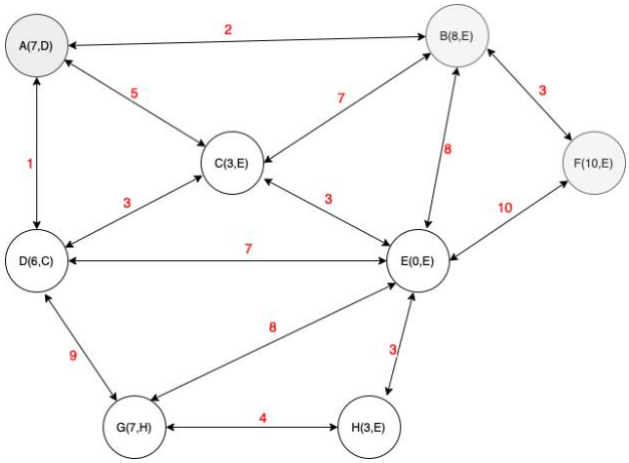
After the operations related to vertex *C* are finished, all remaining neighboring vertices are painted back to gray, the vertex (*H*) with the smallest value among the vertices whose length value is calculated is painted white, and the vertices adjacent to this vertex (*H*) are painted pink and the lengths for these neighboring vertices are calculated. The length value of the *G* is updated because the length value will shorten when the *G* vertex is moved over the *H* vertex.



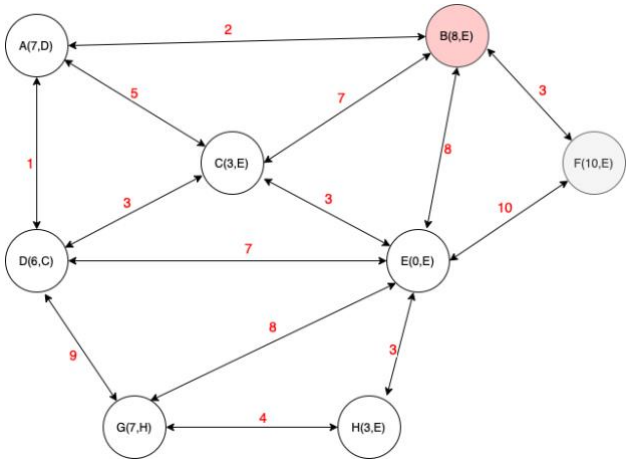
After the operations related to vertex *H* are finished, all remaining neighboring vertices are painted back to gray, the vertex (*D*) with the smallest value among the vertices whose length value is calculated is painted white, and the vertices adjacent to this vertex (*D*) are painted pink and the lengths for these neighboring vertices are calculated. The length value of the *A* is updated because the length value will shorten when the *A* vertex is moved over the *D* vertex.



After the operations related to vertex *D* are finished, all remaining neighboring vertices are painted back to gray, the vertex (*G*) with the smallest value among the vertices whose length value is calculated is painted white, and this vertex (*G*) does not have a gray to white vertex, so no action can be taken.

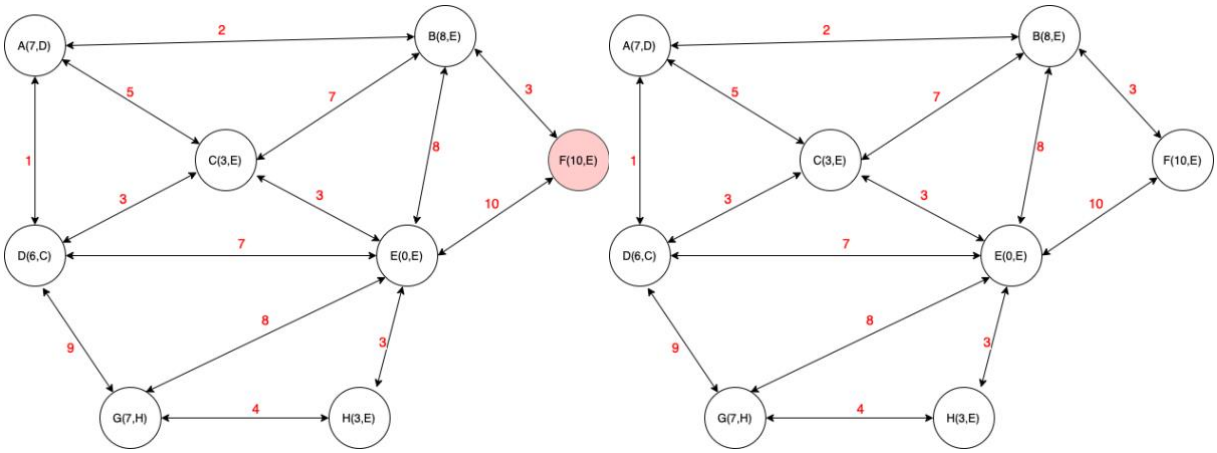


After the operations related to vertex *G* are finished, all remaining neighboring vertices are painted back to gray, the vertex (*A*) with the smallest value among the vertices whose length value is calculated is painted white, and the vertices adjacent to this vertex (*A*) are painted pink and the lengths for these neighboring vertices are calculated. No update is made because the length value of no adjacent vertices is not reduced.

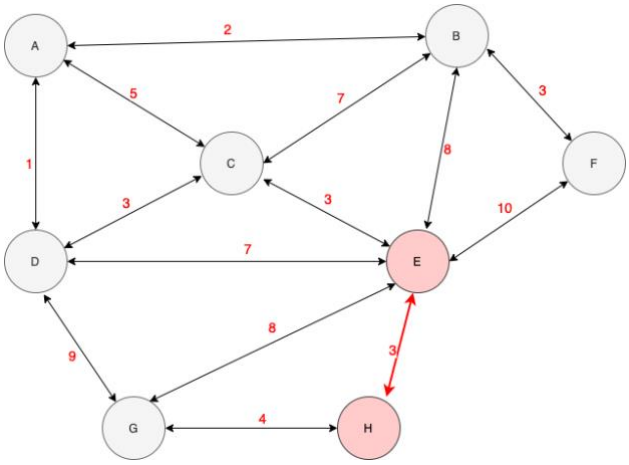


After the operations related to vertex *A* are finished, all remaining neighboring vertices are painted back to gray, the vertex (*B*) with the smallest value among the vertices whose length value is calculated is painted white, and the vertices adjacent to this vertex (*B*) are painted pink and the lengths for these neighboring vertices are calculated. No update is made because the length value of no adjacent vertices is not reduced.

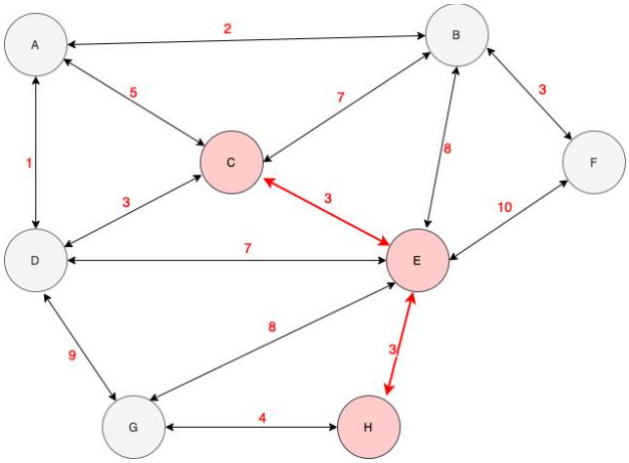
The only remaining vertex (*F*) is painted white, and the process is completed.



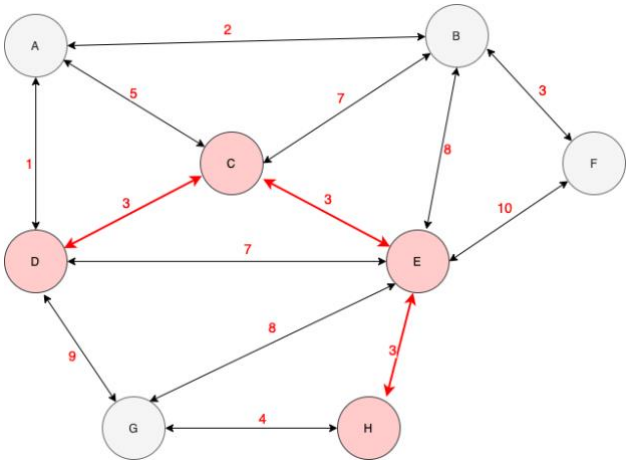
Question 2: Prim's Algorithm is started to be applied from vertex *E*. Vertex *E* is painted in pink and one of the smallest of the edges of the vertices painted in pink is selected and painted together with the vertex (*H*) it connects to. *MST: E, H*



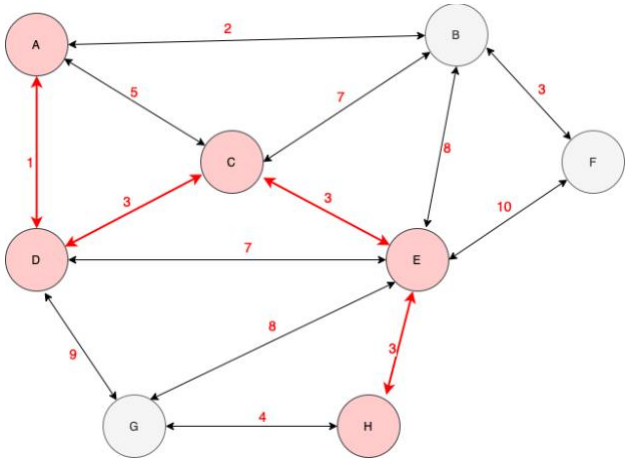
After the *H* vertex is painted pink, one of the smallest of the edges with the other end connected to an unpainted vertex is selected (*C*) and painted together with the vertex to which it is connected. *MST: E, H, C*



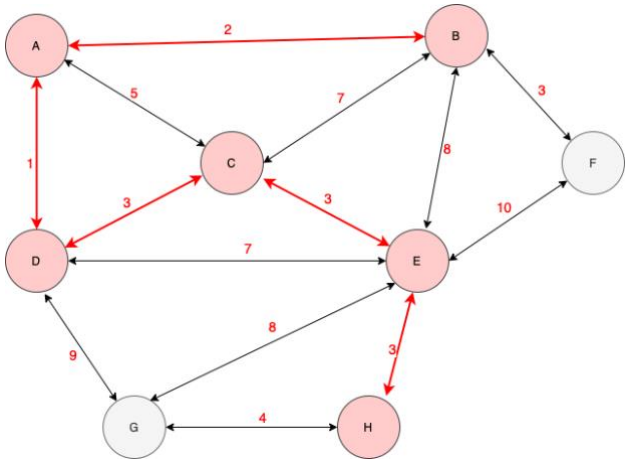
After the *C* vertex is painted pink, one of the smallest of the edges with the other end connected to an unpainted vertex is selected (*D*) and painted together with the vertex to which it is connected. *MST: E, H, C, D*



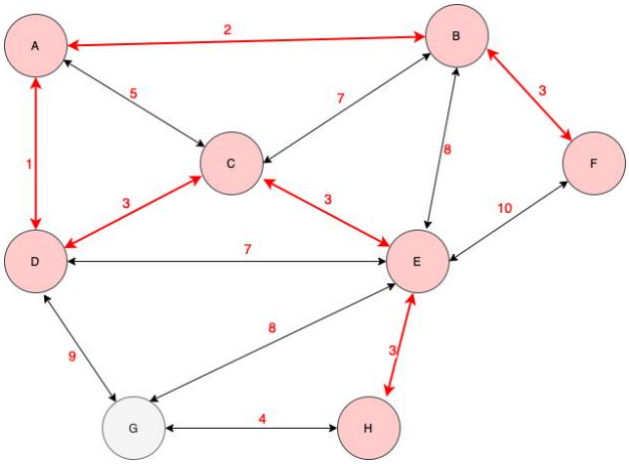
After the *D* vertex is painted pink, one of the smallest of the edges with the other end connected to an unpainted vertex is selected (*A*) and painted together with the vertex to which it is connected. *MST: E, H, C, D, A*



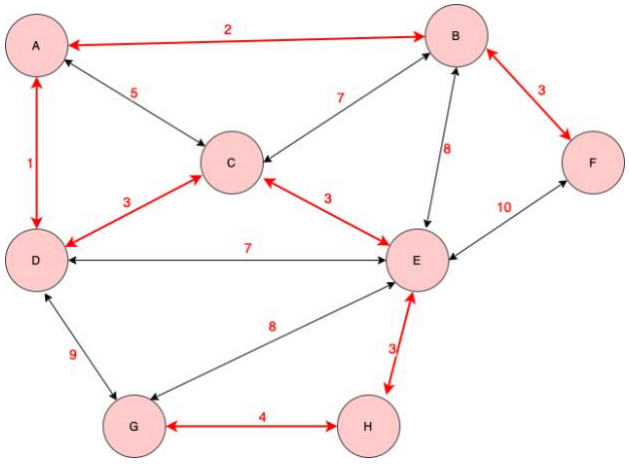
After the *A* vertex is painted pink, one of the smallest of the edges with the other end connected to an unpainted vertex is selected (*B*) and painted together with the vertex to which it is connected. *MST: E, H, C, D, A, B*



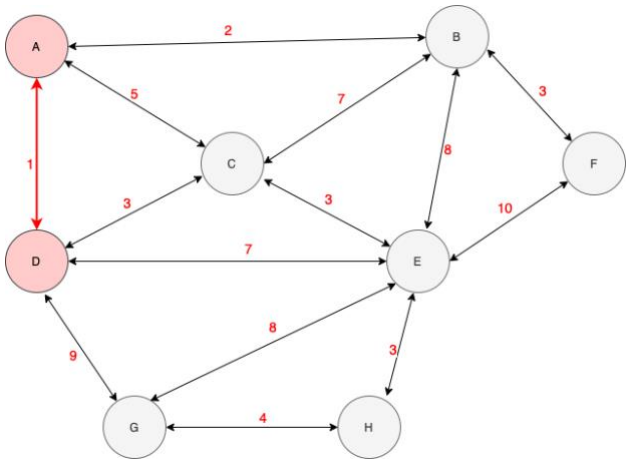
After the *B* vertex is painted pink, one of the smallest of the edges with the other end connected to an unpainted vertex is selected (F) and painted together with the vertex to which it is connected. *MST: E, H, C, D, A, B, F*



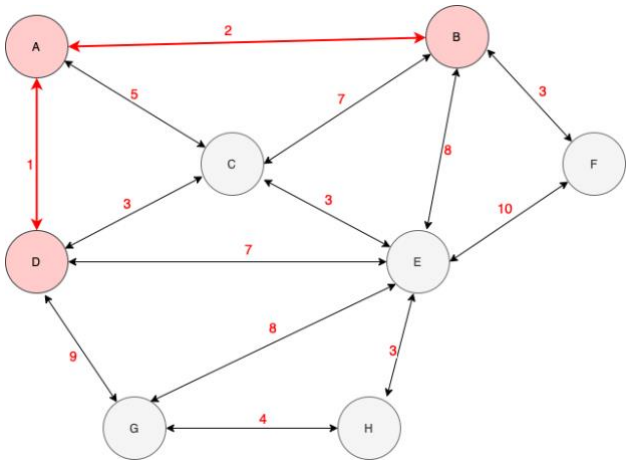
Finally, the boundary from H to G is added to the MST and the MST is completed. *MST: E, H, C, D, A, B, G.*



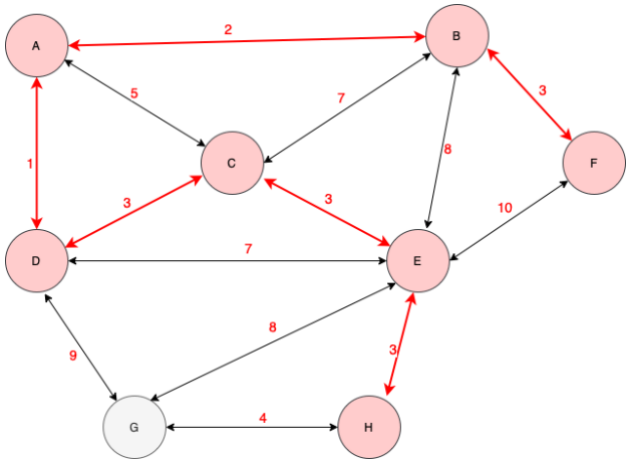
Question 3: According to Kruskal's algorithm, edges are considered from smallest to largest, and at least one of the vertices of the evaluated edge must be unpainted. According to Figure 1, the edge with the lowest length value is between *A-D* and this edge and the vertices to which it is attached are painted.



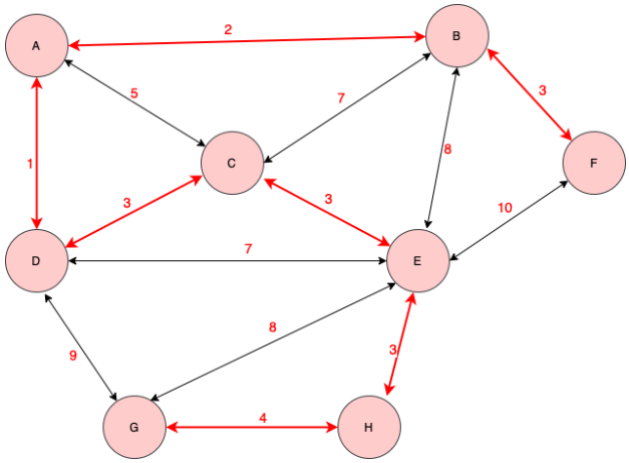
After the border between *A-D* is painted, the border with the smallest length value *(A-B)* and the connected vertices are painted.



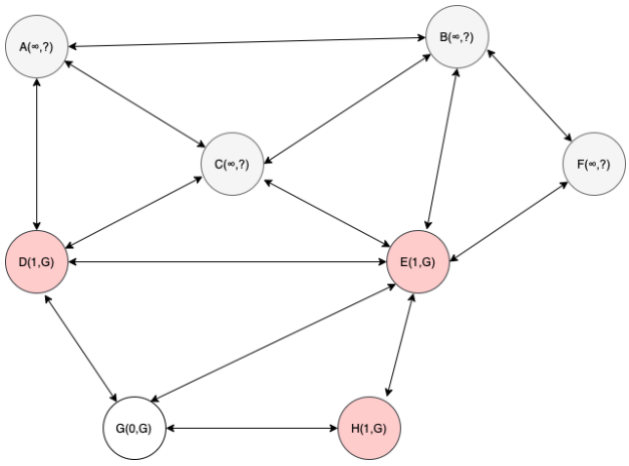
After the edge between *A-B* is painted, the edges with the smallest length values (*B-F, C-D, C-E, E-H*) are painted together with the connected vertices, respectively, since they have at least one unpainted vertex.



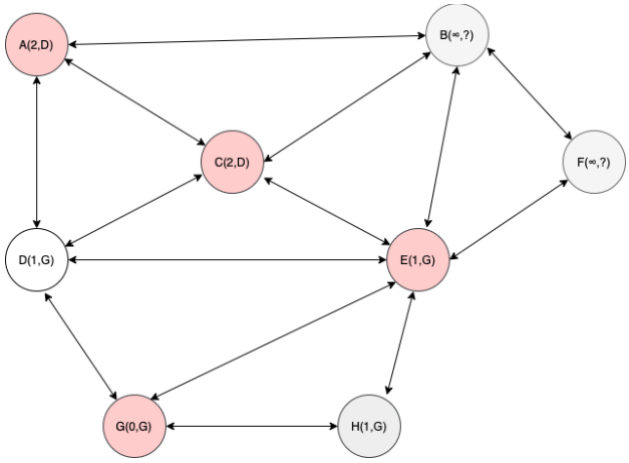
After the *(B-F, C-D, C-E, E-H)* edges are painted respectively, the *G-H* edge with the smallest length value remaining is painted together with the *G* vertex, and the algorithm is completed.



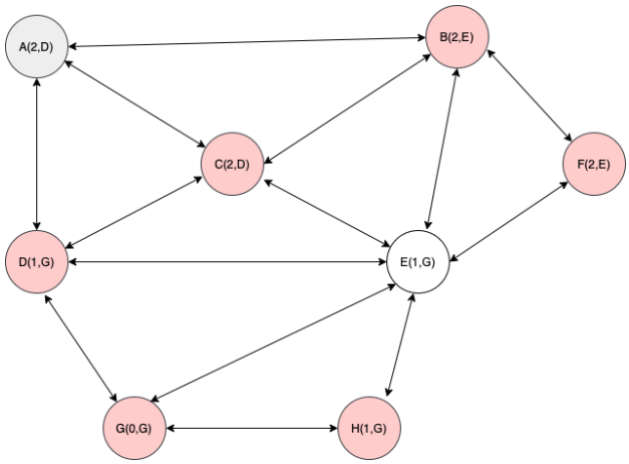
Question 4: According to the BFS algorithm, the starting point of the G vertex is selected, and the adjacent vertices of G are colored pink and the length values are increased by one.



Then, one of the vertices with the smallest length value is selected (*D*) among the visited vertices. If the neighboring vertices of this vertex have never been visited, the value of the vertex they came from is written in these vertices by increasing it by one.



Then, one of the vertices with the smallest length value is selected (*E*) among the visited vertices. If the neighboring vertices of this vertex have never been visited, the value of the vertex they came from is written in these vertices by increasing it by one.



After all the vertices are visited, the algorithm is completed and since it is specified which vertex is reached when coming to a corner, it can be found which way to go from the beginning to a vertex.

Question 5: First, an unreachable vertex is selected from any vertex and printed. Then this vertex is removed from the graph and the same operations are continued until there is no vertex left in the graph. Order: S – G – D – A – B – H – E – I – F – C – T