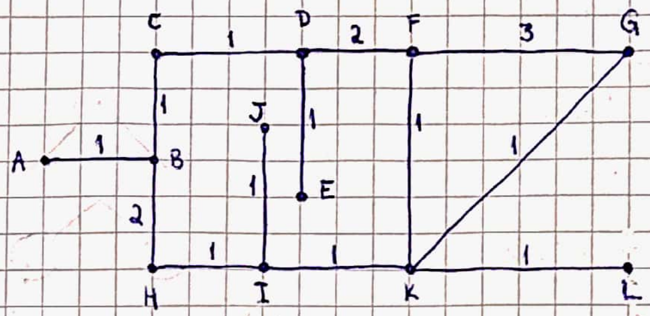
PROBLEM SET 3: Graphs and trees.



EXERCISE 1. (Find the path from node A to node G, making use of the Dijkstra's Algorithm).

	L_0	L_1	L_2	L_3	L_4	L_5	L_6	L_7	L_8	L_9	L_{10}	L_{11}
A	(0)	-	-	-	-	-	-	-	-	-	-	-
B	∞	(1_A)	-	-	-	-	-	-	-	-	-	-
C	∞	∞	(2_B)	-	-	-	-	-	-	-	-	-
D	∞	∞	∞	(3_C)	-	-	-	-	-	-	-	-
E	∞	∞	∞	∞	4_D	(4_D)	-	-	-	-	-	-
F	∞	∞	∞	∞	5_D	5_D	5_D	(5_D)	-	-	-	-
G	∞	∞	∞	∞	∞	∞	∞	∞	8_F	8_F	(6_K)	
H	∞	∞	3_B	3_B	(3_B)	-	-	-	-	-	-	-
I	∞	∞	∞	∞	∞	4_H	(4_H)	-	-	-	-	-
J	∞	∞	∞	∞	∞	∞	∞	5_I	(5_I)	-	-	-
K	∞	∞	∞	∞	∞	∞	∞	5_I	5_I	(5_I)	-	-
L	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	6_I	6_I

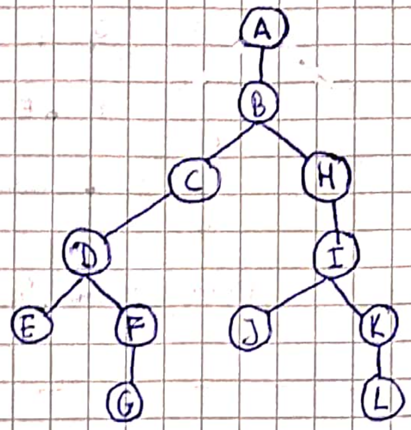
When G is selected, the alg. ends

Weight of the optimal path: 6.

Optimal path: A → B → H → I → K → G

1 + 2 + 1 + 1 + 1 = 6.

EXERCISE 2. Obtain the Breadth-first-search tree corresponding to the graph, starting from node A.



(Using the BFS alg.)

Source: $s = A$. $Q = \{A_0\}$

expand A: $Q = \{B_1\}$

expand B: $Q = \{C_2, H_2\}$

expand C: $Q = \{H_2, D_3\}$

expand H: $Q = \{D_3, I_3\}$

expand D: $Q = \{I_3, E_4, F_4\}$

expand I: $Q = \{E_4, F_4, J_4, K_4\}$

expand E: $Q = \{F_4, J_4, K_4\}$

expand F: $Q = \{J_4, K_4, G_5\}$

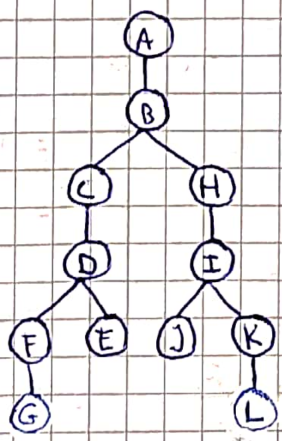
expand J: $Q = \{K_4, G_5\}$

expand K: $Q = \{G_5, L_5\}$

expand G: $Q = \{L_5\}$

expand L: $Q = \emptyset \to$ end.

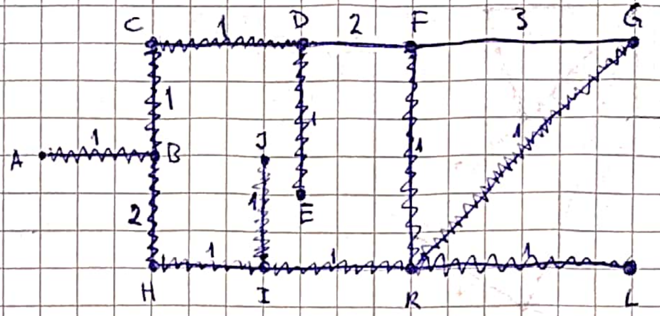
BFS tree:

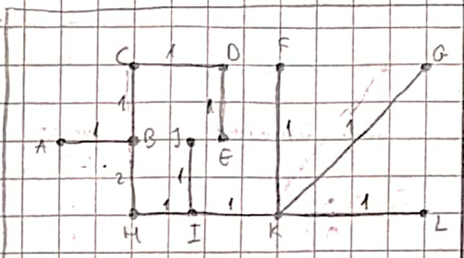


Scanned by CamScanner

EXERCISE 3:

(Prim's algorithm)





Minimum weight spanning tree.

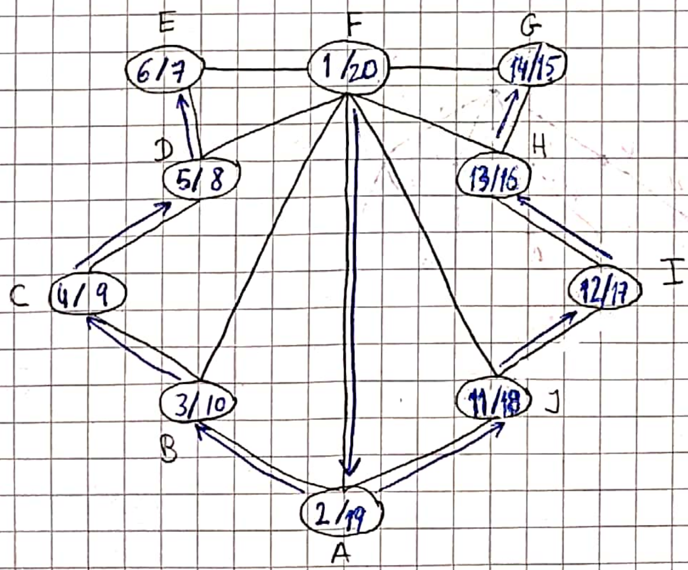
Edge considered	Weight	Select?
AB	1	yes.
BC	1	yes
CD	1	yes
DE	1	yes
BH	2	yes
HI	1	yes
IJ	1	yes
IK	1	yes
FK	1	yes
GK	1	yes
KL	1	yes
DF	(2)	no
FG	(3)	no

not selected because they would form a cycle.

Total weight of the minimum cost spanning tree: 12.

EXERCISE 4

DFS: starting from node F (following the DFS algorithm)



EXERCISE 5: Find the optimal (lowest cost) path to *Santiago de Compostela* from **node a** to **node g**, making use of Dijkstra's algorithm. Detail each of the steps of the algorithm in the table below, and indicate the optimal path and its weight. Use as many columns and rows in the table as necessary.

	L_0	L_1	L_2	L_3	L_4	L_5	L_6	L_7	
A	(0)	–	–	–	–	–	–	–	
B	∞	(1_A)	–	–	–	–	–	–	
C	∞	∞	(2_B)	–	–	–	–	–	
D	∞	∞	∞	6_C	(5_F)	–	–	–	
E	∞	∞	∞	∞	24_F	24_F	24_F	24_F	
F	∞	20_A	4_B	(4_B)	–	–	–	–	
G	∞	∞	∞	∞	44_F	44_F	6_H	(6_H)	
H	∞	∞	∞	∞	5_F	(5_F)	–	–	
I	∞	∞	∞	∞	∞	∞	6_H	6_H	
J	∞	20_A	20_A	20_A	5_F	5_F	(5_F)	–	

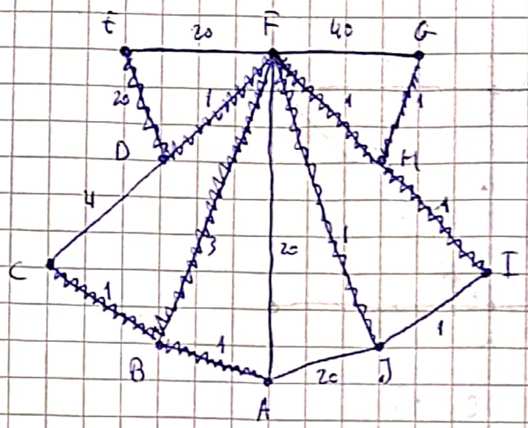
G is selected, so the procedure ends.

Optimal path: $A \rightarrow B \rightarrow F \rightarrow H \rightarrow G$

$1 + 3 + 1 + 1 = 6.$

Weight of the optimal path: 6.

EXERCISE 6 Kruskal's algorithm is asked.



Edge considered	Weight	Selected?
AB	1	yes
BC	1	yes
DF	1	yes
FH	1	yes
FJ	1	yes
GH	1	yes
HI	1	yes
IJ	(1)	no → it forms a cycle
BF	3	yes
CD	(4)	no
DE	20	yes → we stop here, because we have already selected n-1 edges, where n = # nodes = 10.

Weight of the minimum cost spanning tree: 30.

Spanning tree: