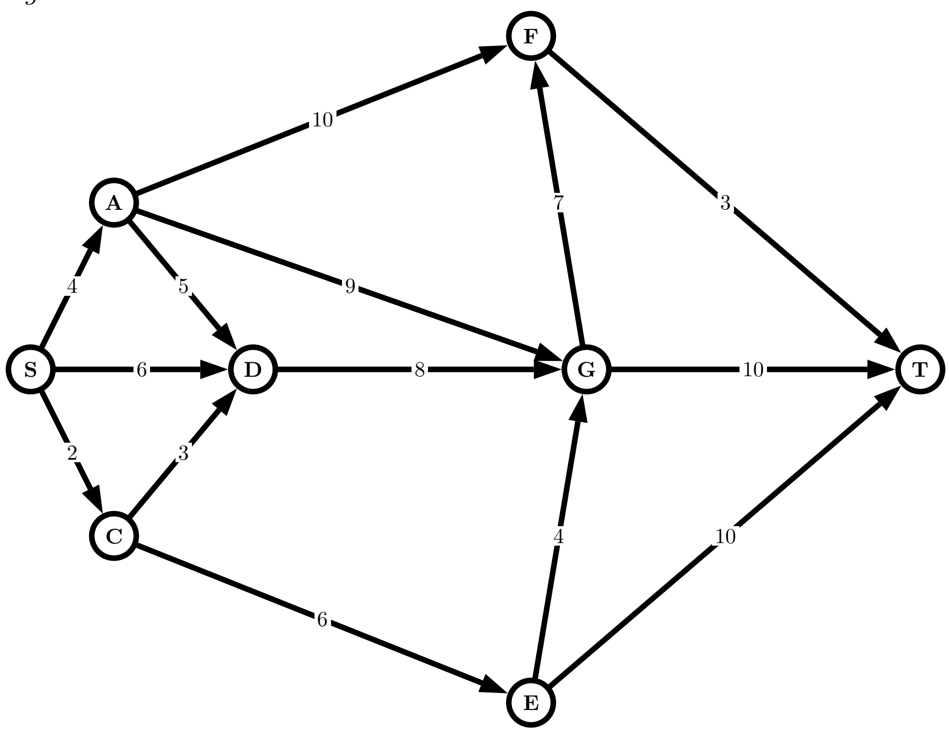
Mathematics for Computer Science, CM0167,
Example class, Week 6,

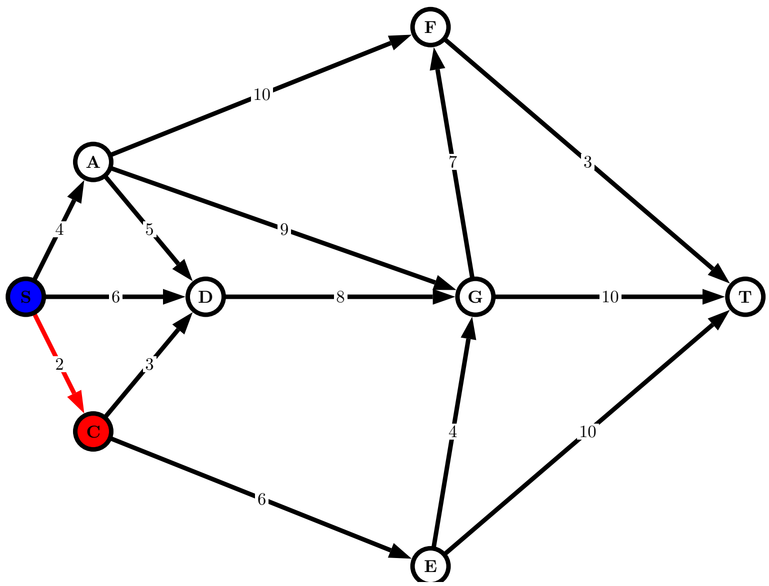
SOLUTIONS
Dr David Marshall

1. *Find a minimum spanning tree for the weighted graph below using Prim's algorithm.*

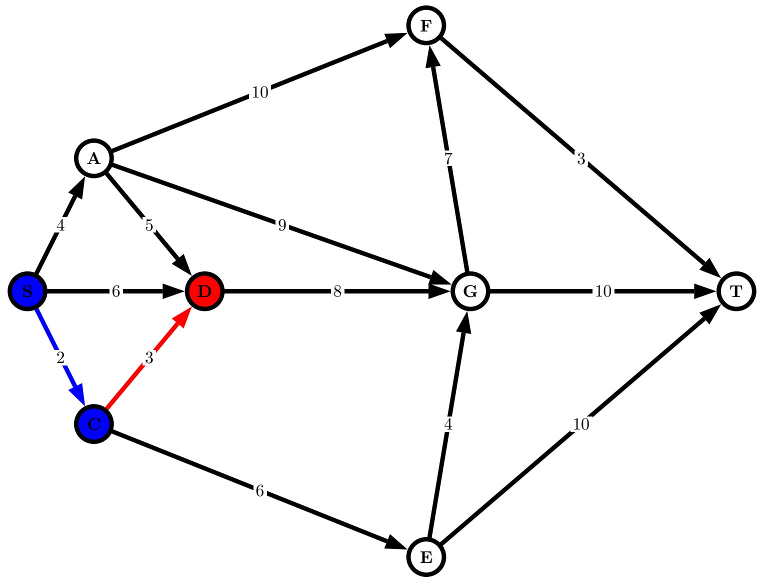


Apply Prim's algorithm as follows:

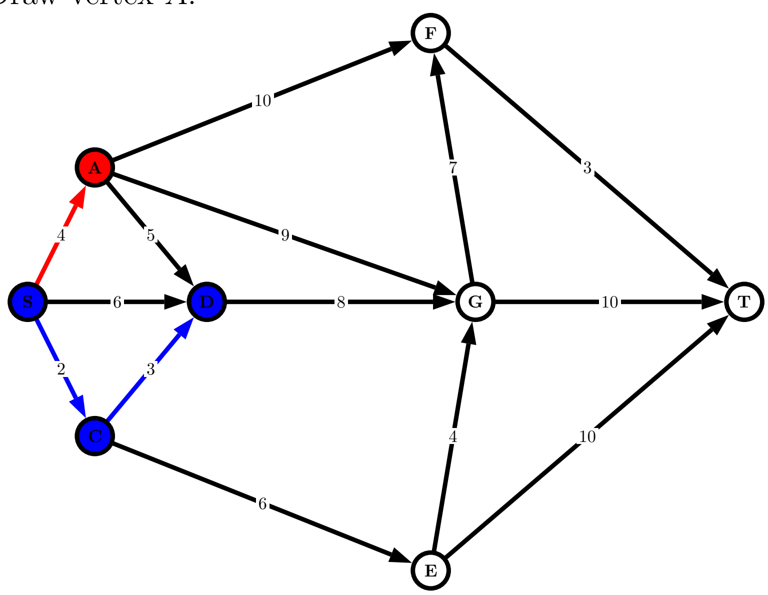
(a) Start: Choose S, Draw S
Lowest weight is SC choose this path and draw it. Draw Vertex C.


F
A
10
3
7
4
5
9
S
6
D
8
G
10
T
2
3
C
4
10
6
E

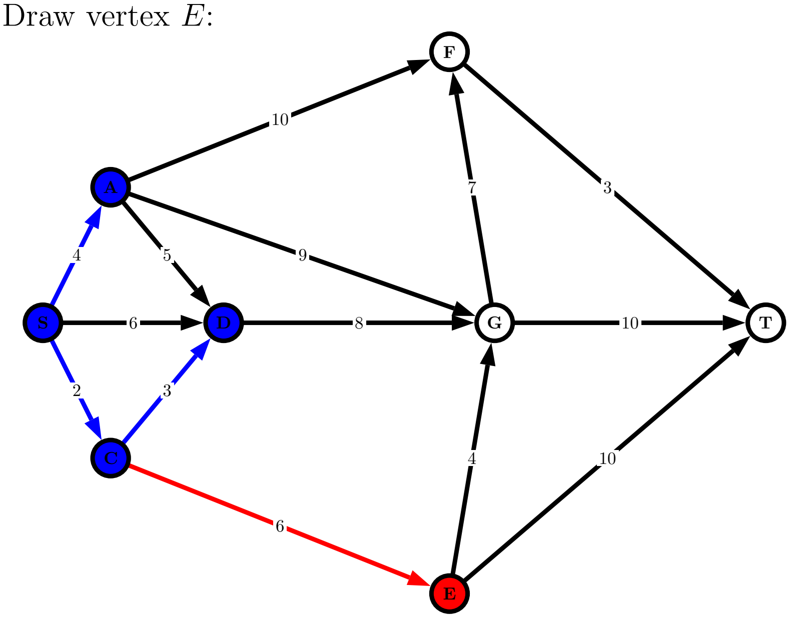
(b) Vertices S and C now drawn. Lowest weighed edge between drawn vertices and non-drawn is CD choose this path and draw it, Draw vertex D:



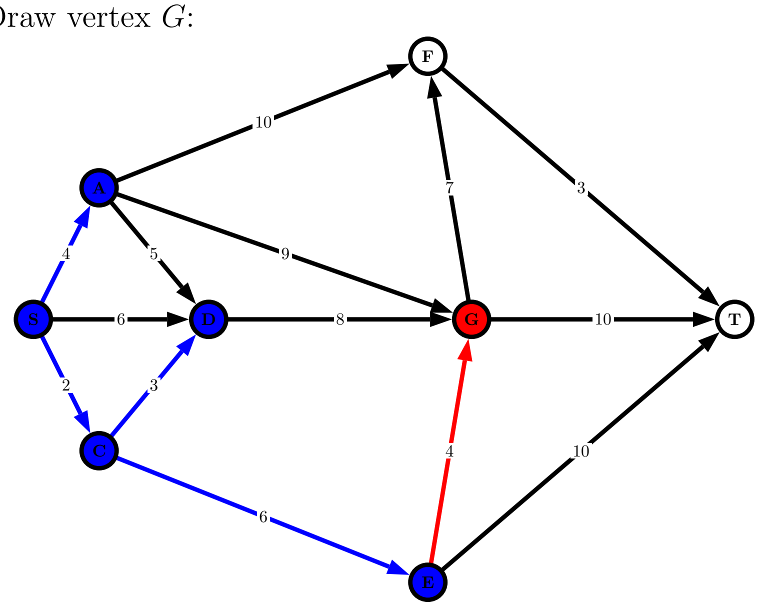
(c) Vertices S, C and D now drawn. Lowest weighed edge between drawn vertices and non-drawn is SA choose this path and draw it, Draw vertex A:



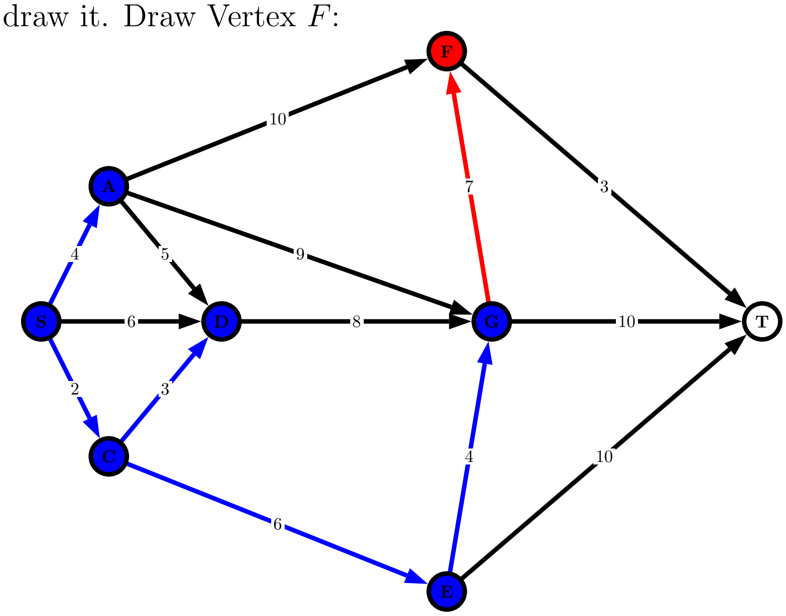
(d) Vertices S, C, D and A now drawn. Lowest weighed edge between drawn vertices and non-drawn is CE choose this path and draw it, Draw vertex E:



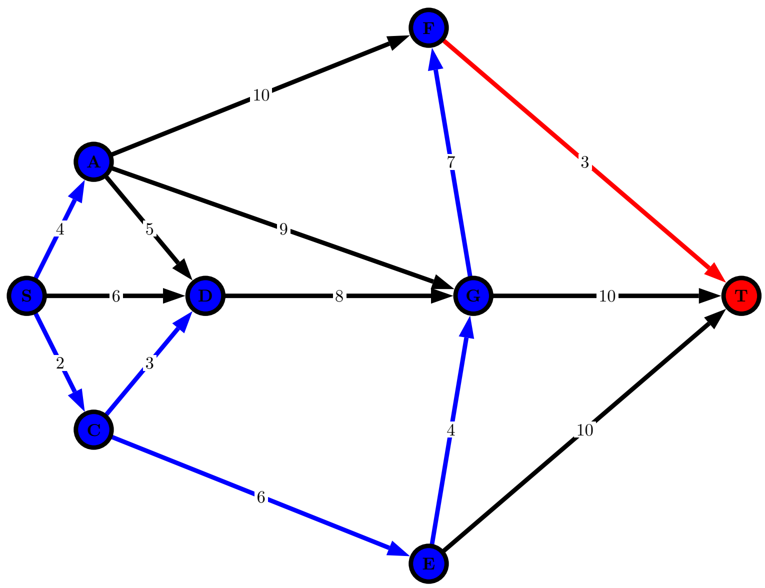
(e) Vertices S, C, D, A and E now drawn. Lowest weighed edge between drawn vertices and non-drawn is EG choose this path and draw it, Draw vertex G:



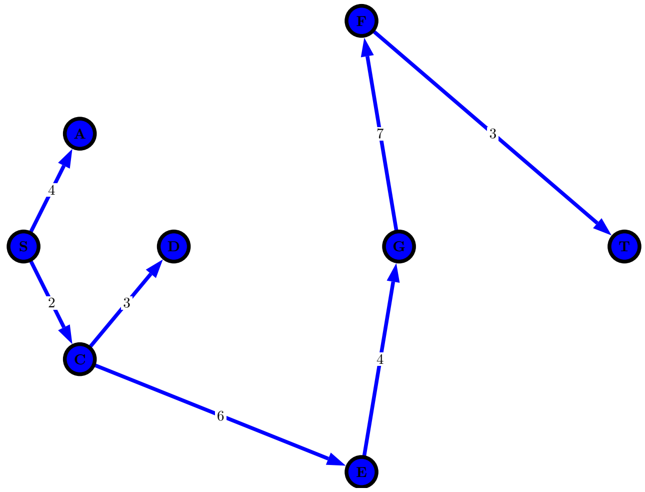
(f) Vertices S, C, D, A, E and G now drawn. Lowest weighed edge between drawn vertices and non-drawn is FG choose this path and draw it. Draw Vertex F:



(g) Vertices S, C, D, A, E, G and F now drawn (Only T left). Lowest weighed edge between drawn vertices and non-drawn is FT choose this path and draw it. Draw Vertex T:



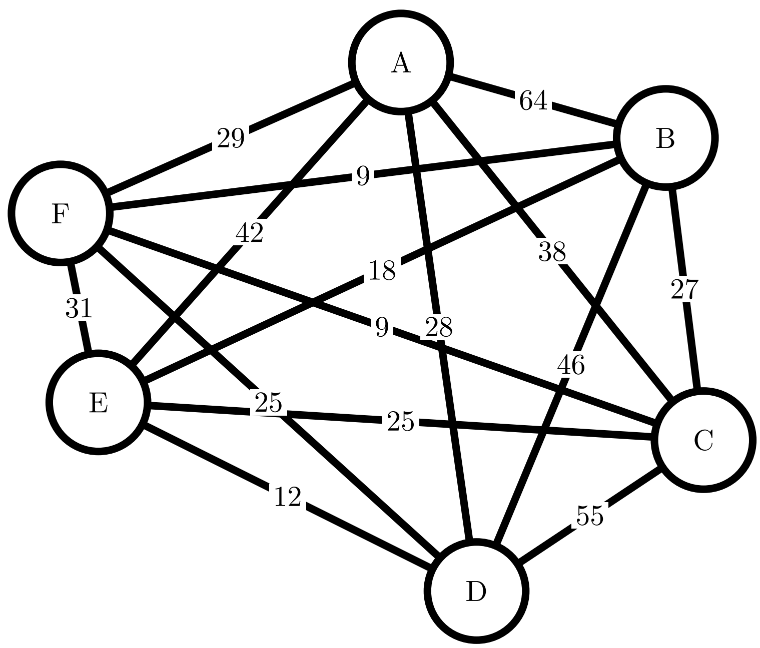
So the minimum spanning tree is:



2. *Find an upper and lower bound for the* Travelling Salesperson Problem *for the cities A, B, C, D, E and F.*

	A	B	C	D	E	F
A	–	64	38	28	42	29
B	64	–	27	46	18	9
C	38	27	–	55	25	9
D	28	46	55	–	12	25
E	42	18	25	12	–	31
F	29	9	9	25	31	–

Graph for above data is:



(a) Upper Bound Solution

Recap: *listing of algorithm not required for solution* To get an upper bound we use the following algorithm (The heuristic/nearest neighbour algorithm)

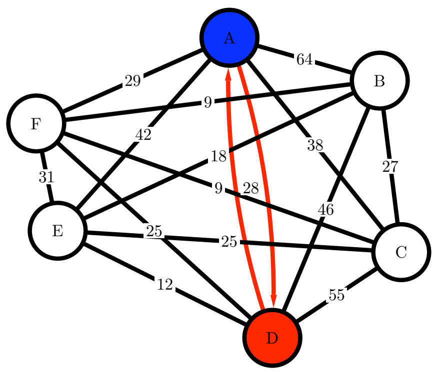
The idea for the heuristic algorithm is similar to the idea of Prim's algorithm, except that we build up a cycle rather than a tree.

- START with all the vertices of a complete weighted graph.
- Step 1: Choose any vertex and find a vertex joined to it by an edge of minimum weight. Draw these two vertices and join them with two edges to form a cycle. Give the cycle a clockwise rotation.

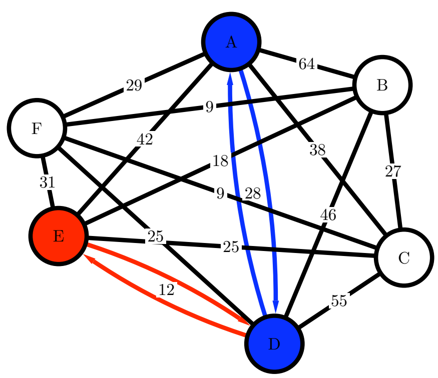
- Step 2: Find a vertex not currently drawn, joined by an edge of least weight to a vertex already drawn. Insert this new vertex into the cycle in front of the 'nearest' already connected vertex.
- REPEAT Step 2 until all the vertices are joined by a cycle, then STOP.

The total weight of the resulting Hamiltonian cycle is then an upper bound for the solution to the travelling salesperson problem.

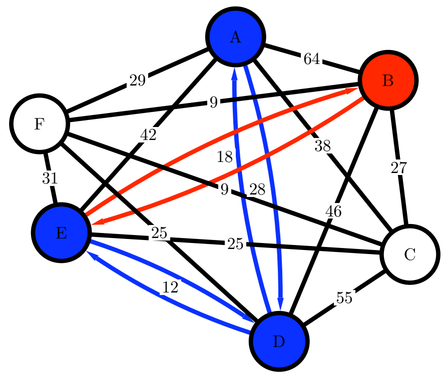
Step 1: Choose Vertex A (*if other vertices chosen a valid but different answer possible*). Draw A. Lowest weight is AD. So Draw D and draw AD as a clockwise cycle.



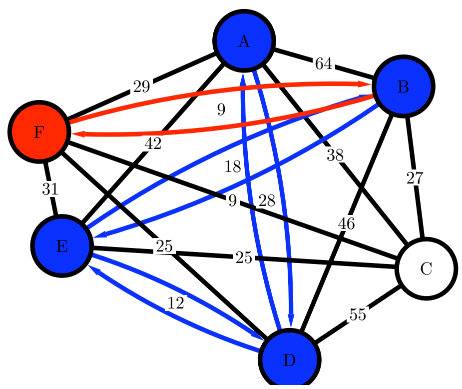
Step 2: Vertices Drawn: A, D. Lowest Weight from undrawn vertex to drawn vertex is DE. So Draw E and draw DE as a clockwise cycle.



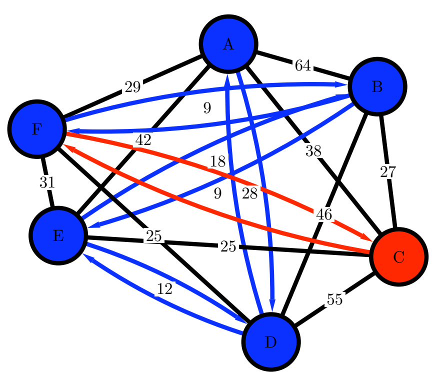
Step 3: Vertices Drawn: A, D, E. Lowest Weight from undrawn vertex to drawn vertex is EB. So Draw B and draw EB as a clockwise cycle.



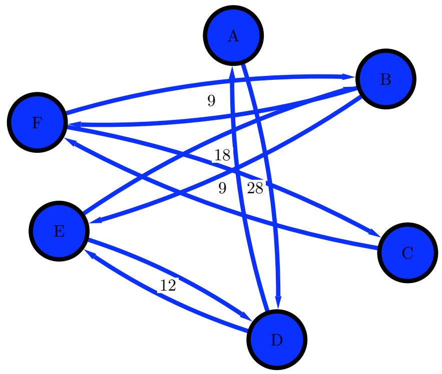
Step 4: Vertices Drawn: A, D, E, B. Lowest Weight from undrawn vertex to drawn vertex is BF. So Draw F and draw BF as a clockwise cycle.



Step 5: Vertices Drawn: A, D, E, B, F. Lowest Weight from undrawn vertex to drawn vertex is FC. So Draw C and draw FC as a clockwise cycle.



So Hamiltonian Cycle create is:



The **Upper BOUND** for the TSP of this problem is the weight of this cycle which is:

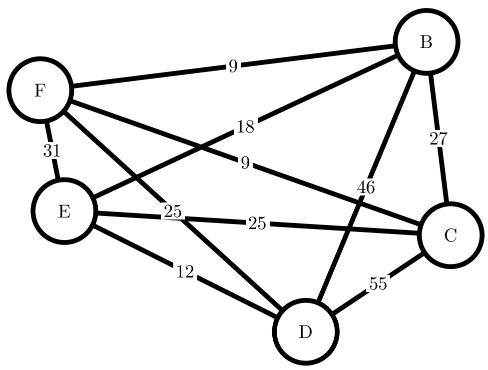
$$2 \times (28 + 12 + 18 + 9 + 9) = 152$$

(b) Lower Bound Solution

Recap: *listing of algorithm not required for solution* Lower bound for the travelling salesperson problem algorithm:

- Step 1: Choose a vertex V and remove it from the graph.
- Step 2: Find a minimum spanning tree connecting the remaining vertices, and calculate its total weight w.
- Step 3: Find the two smallest weights, w_1 and w_2, of edges incident with V.
- Step 4: Calculate the lower bound $w + w_1 + w_2$.

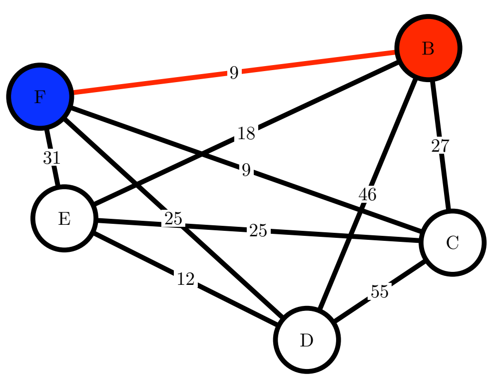
Step 1: Choose Vertex A. Remove A from the graph:



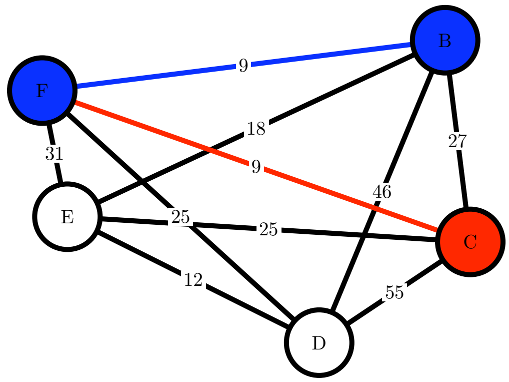
Step 2: Use Prim's Algorithm to find minimum spanning tree:

- START with all the vertices of a weighted graph.
- Step 1: Choose and draw any vertex.
- Step 2: Find the edge of least weight joining a drawn vertex to a vertex not currently drawn. Draw this weighted edge and the corresponding new vertex .
- REPEAT Step 2 until all the vertices are connected, then STOP.

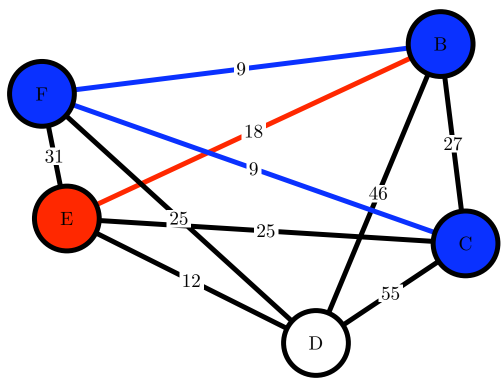
Step 2.1: Choose and draw F. Edge of least weight to F is either B or C. Choose B. Draw B and edge FB:



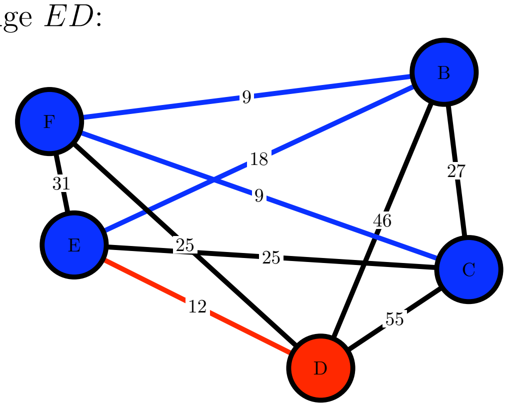
Step 2.2: Vertices drawn: F, B . Edge of least weight from a non-drawn vertex to a drawn vertex is FC Choose C. Draw C and edge FC:



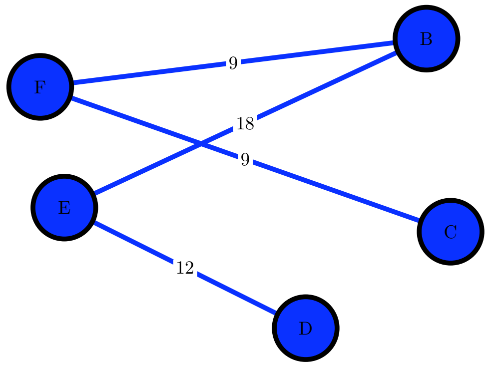
Step 2.3: Vertices drawn: F, B, C. Edge of least weight from a non-drawn vertex to a drawn vertex is BE Choose E. Draw E and edge BE:



Step 2.3: Vertices drawn: F, B, C, E. Edge of least weight from a non-drawn vertex to a drawn vertex is ED Choose D. Draw D and edge ED:



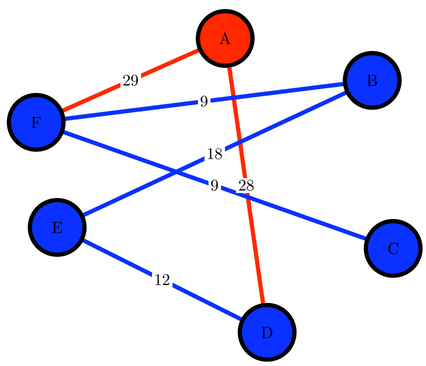
Step 2.4: So minimum spanning tree is:



The weight of this tree is:

$$w = 9 + 9 + 18 + 12 = 48$$

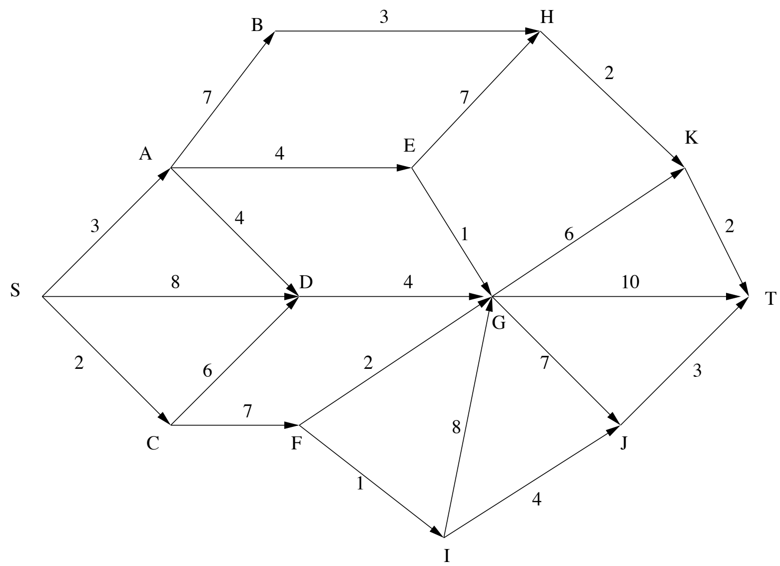
Step 3 : Now add the two least weighted edges to A which are AD with weight $w_1 = 28$ and AF with weight $w_2 = 29$



So the **lower bound** of this TSP problem is:

$$w + w_1 + w_2 = 48 + 28 + 29 = 105$$

3. *Find the shortest path from S to T in the digraph below using Dijkstra's algorithm. Show your working with tables.*



Applying Dijkstra's algorithm we get the following table for the route from S to T:

Step	Vertex marked	Current potential	Distance to Vertex													Unchosen vertices
			S	A	B	C	D	E	F	G	H	I	J	K	T	
1	S	0	0	3	–	**2**	8	–	–	–	–	–	–	–	–	A,B,C,D,E,F,G,H,I,J,K,T
2	C	2	0	**3**	–	**2**	8	–	9	–	–	–	–	–	–	A,B,D,E,F,G,H,I,J,K,T
3	A	3	0	**3**	10	**2**	**7**	7	9	–	–	–	–	–	–	B,D,E,F,G,H,I,J,K,T
4	D	7	0	**3**	10	**2**	**7**	**7**	9	11	–	–	–	–	–	B,E,F,G,H,I,J,K,T
5	E	7	0	**3**	10	**2**	**7**	**7**	9	**8**	14	–	–	–	–	B,F,G,H,I,J,K,T
6	G	8	0	**3**	10	**2**	**7**	**7**	**9**	**8**	14	–	15	14	18	B,F,H,I,J,K,T
7	F	9	0	**3**	**10**	**2**	**7**	**7**	**9**	**8**	14	10	15	14	18	B,H,I,J,K,T
8	B	10	0	**3**	**10**	**2**	**7**	**7**	**9**	**8**	13	**10**	15	14	18	H,I,J,K,T
9	I	10	0	**3**	**10**	**2**	**7**	**7**	**9**	**8**	**13**	**10**	14	14	18	H,J,K,T
10	H	13	0	**3**	**10**	**2**	**7**	**7**	**9**	**8**	**13**	**10**	**14**	14	18	J,K,T
11	J	14	0	**3**	**10**	**2**	**7**	**7**	**9**	**8**	**13**	**10**	**14**	**14**	17	K,T
12	K	14	0	**3**	**10**	**2**	**7**	**7**	**9**	**8**	**13**	**10**	**14**	**14**	16	T

This is explained as follows (**= choose**, **= chosen**, **= dont overwrite**):

Step 1 — Only Valid Paths from S are to vertices A, B and D. Add weights in table choose lowest with is **colorblue** C

Step 2 — Update Current Potential (2). C can link to D, potential 8 so can update as its equal to current potential. and F which is new. A is lowest choose this.

Step 3 — Update Current Potential (3). A can link to D, potential 7 so can update as its less than current potential (8). and F which is new. A is lowest choose this. A links to B and also E which are new. D and E joint lowest. Let's choose D.

Step 4 — Update Current Potential (7). D can link to G, which is new. E is lowest. Choose E.

Step 5 — Update Current Potential (7). E can link to H, which is new, and G, which is a lower potential than current so replace. G is lowest. Choose G.

Step 6 — Update Current Potential (8). G can link to K,T and J, which are new. F is lowest. Choose F.

Step 7 — Update Current Potential (9). Can Link to G but its a higher potential. G connects to I which is new and the lowest potential. Choose B.

Step 8 — Update Current Potential (10). Can Link to H and its has a lower potential so update. Choose I.

Step 9 — Update Current Potential (10). Can Link to G but its a higher potential. I connect to J which is lower than current potential. Choose H.

Step 10 — Update Current Potential (13). Can Link to K but its a higher potential. Choose J as it has lowest potential.

Step 11 — Update Current Potential (14). Can Link to T so update as this is a lower potential so update. Choose K as it has lowest potential.

Step 12 — Update Current Potential (14). Can Link to T but this is higher than current.

So lowest weight/shortest path from S to T is $SAEGKT$ with a weight of 16.