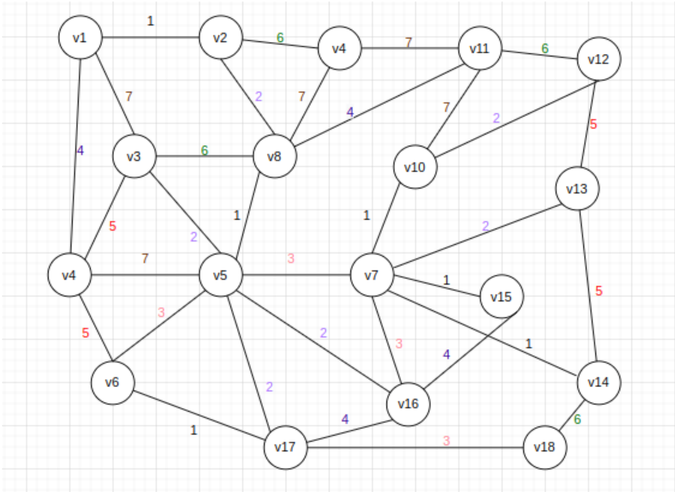
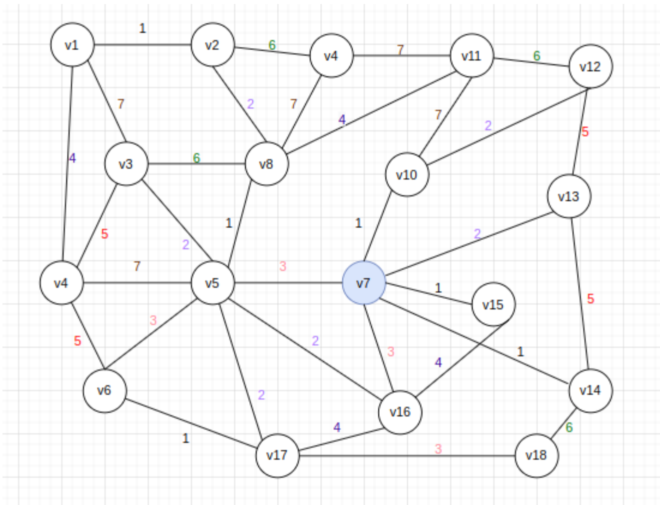
1. Proporcionar una gráfica conexa G = (V, A) con al menos 18 vértices y al menos 36 aristas con pesos positivos en el intervalo $[1, 7] \subset \mathbf{Z}$; deber a haber al menos cuatro aristas con cada costo c, $c \in [1, 7] \subset \mathbf{Z}$.
Aplicar los siguientes algoritmos a la gráfica dada G, ilustrando cómo van transformándose las estructuras y mostrando al final los valores de las etiquetas para cada vértice.
a) Prim usando Heaps Binarios.

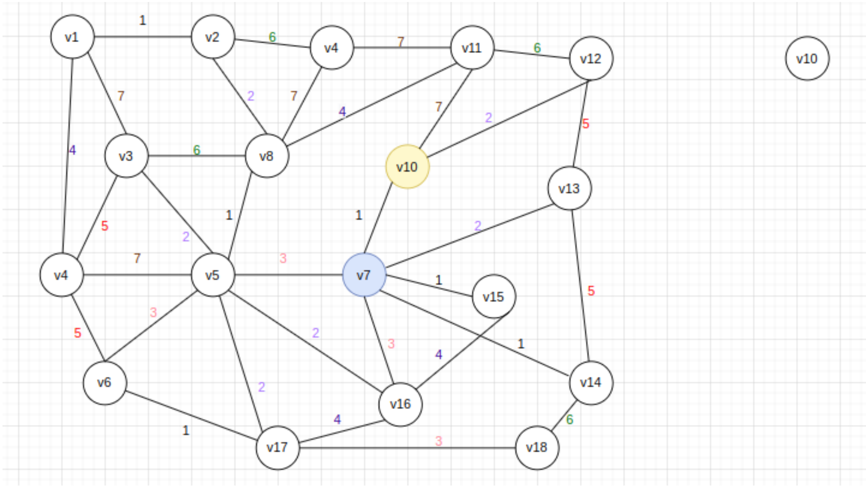


Escogemos el Vertice **v7** para empezar el algoritmo

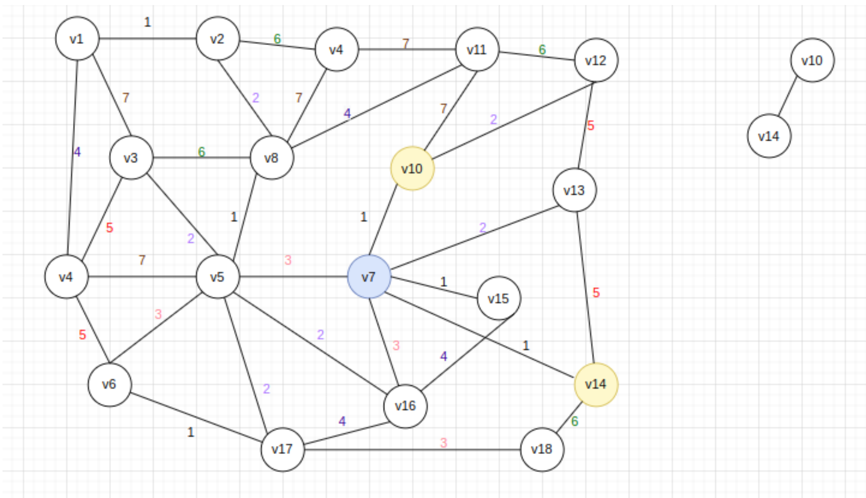
Ver	Pad	Cos	Vis	Ite
v_1	x	∞	F	F
v_2	x	∞	F	F
v_3	x	∞	F	F
v_4	x	∞	F	F
v_5	x	∞	F	F
v_6	x	∞	F	F
v_7	x	∞	T	T
v_8	x	∞	F	F
v_9	x	∞	F	F
v_{10}	x	∞	F	F
v_{11}	x	∞	F	F
v_{12}	x	∞	F	F
v_{13}	x	∞	F	F
v_{14}	x	∞	F	F
v_{15}	x	∞	F	F
v_{16}	x	∞	F	F
v_{17}	x	∞	F	F
v_{18}	x	∞	F	F



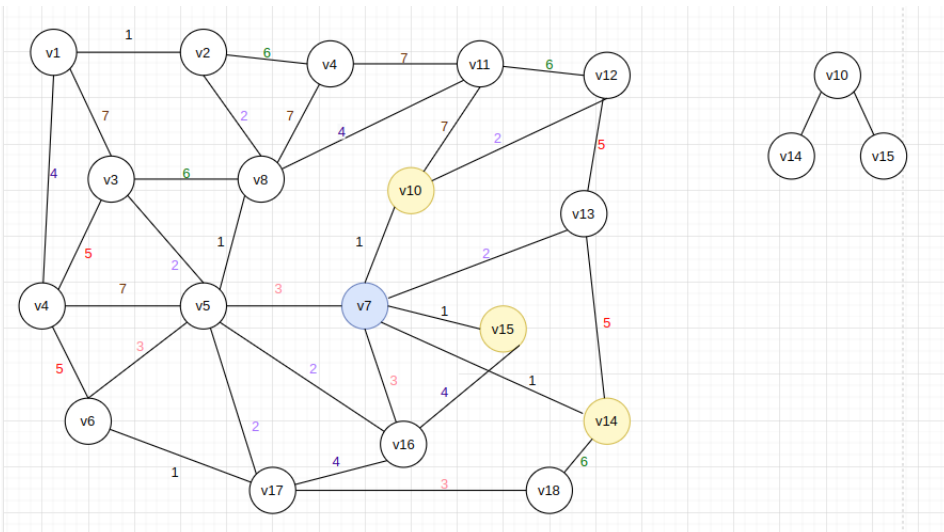
Ver	Pad	Cos	Vis	Ite
v_1	x	∞	F	F
v_2	x	∞	F	F
v_3	x	∞	F	F
v_4	x	∞	F	F
v_5	x	∞	F	F
v_6	x	∞	F	F
v_7	x	∞	T	T
v_8	x	∞	F	F
v_9	x	∞	F	F
v_{10}	v_7	1	T	F
v_{11}	x	∞	F	F
v_{12}	x	∞	F	F
v_{13}	x	∞	F	F
v_{14}	x	∞	F	F
v_{15}	x	∞	F	F
v_{16}	x	∞	F	F
v_{17}	x	∞	F	F
v_{18}	x	∞	F	F



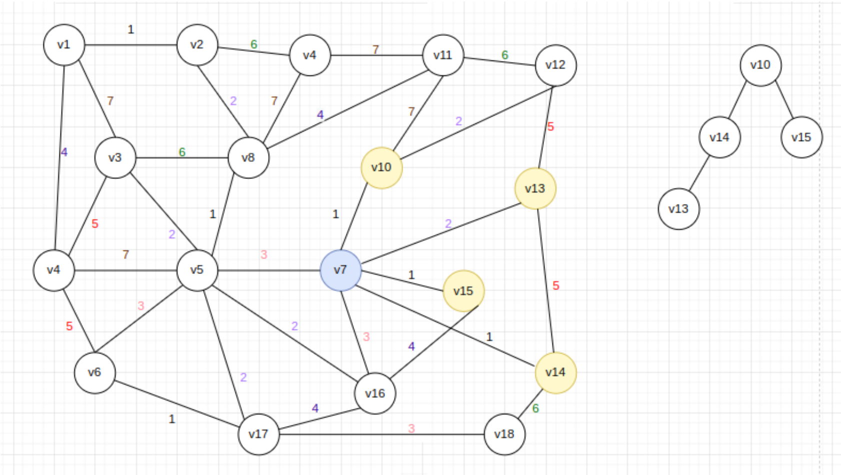
Ver	Pad	Cos	Vis	Ite
v_1	x	∞	F	F
v_2	x	∞	F	F
v_3	x	∞	F	F
v_4	x	∞	F	F
v_5	x	∞	F	F
v_6	x	∞	F	F
v_7	x	∞	T	T
v_8	x	∞	F	F
v_9	x	∞	F	F
v_{10}	v_7	1	T	F
v_{11}	x	∞	F	F
v_{12}	x	∞	F	F
v_{13}	x	∞	F	F
v_{14}	v_7	1	T	F
v_{15}	x	∞	F	F
v_{16}	x	∞	F	F
v_{17}	x	∞	F	F
v_{18}	x	∞	F	F



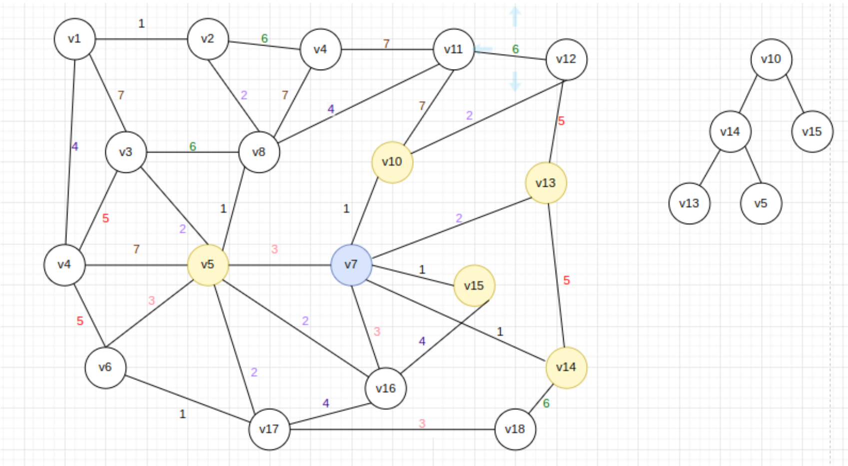
Ver	Pad	Cos	Vis	Ite
v_1	x	∞	F	F
v_2	x	∞	F	F
v_3	x	∞	F	F
v_4	x	∞	F	F
v_5	x	∞	F	F
v_6	x	∞	F	F
v_7	x	∞	T	T
v_8	x	∞	F	F
v_9	x	∞	F	F
v_{10}	v_7	1	T	F
v_{11}	x	∞	F	F
v_{12}	x	∞	F	F
v_{13}	x	∞	F	F
v_{14}	v_7	1	T	F
v_{15}	v_7	1	T	F
v_{16}	x	∞	F	F
v_{17}	x	∞	F	F
v_{18}	x	∞	F	F



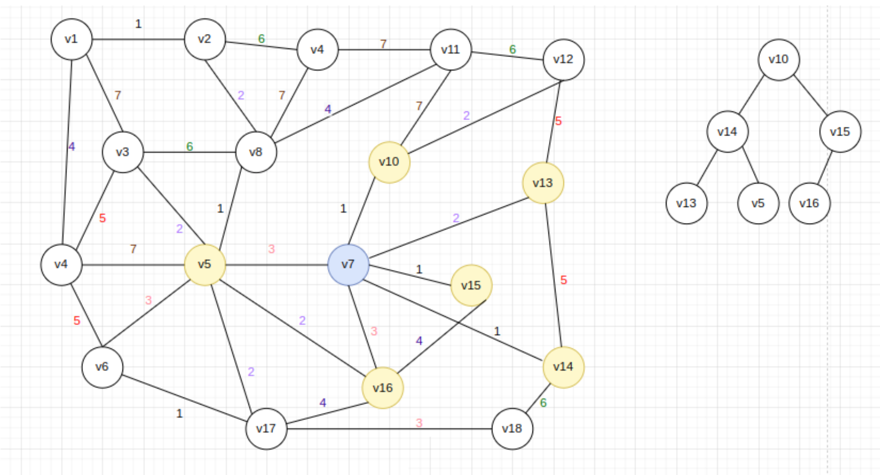
Ver	Pad	Cos	Vis	Ite
v_1	x	∞	F	F
v_2	x	∞	F	F
v_3	x	∞	F	F
v_4	x	∞	F	F
v_5	x	∞	F	F
v_6	x	∞	F	F
v_7	x	∞	T	T
v_8	x	∞	F	F
v_9	x	∞	F	F
v_{10}	v_7	1	T	F
v_{11}	x	∞	F	F
v_{12}	x	∞	F	F
v_{13}	v_7	2	T	F
v_{14}	v_7	1	T	F
v_{15}	v_7	1	T	F
v_{16}	x	∞	F	F
v_{17}	x	∞	F	F
v_{18}	x	∞	F	F



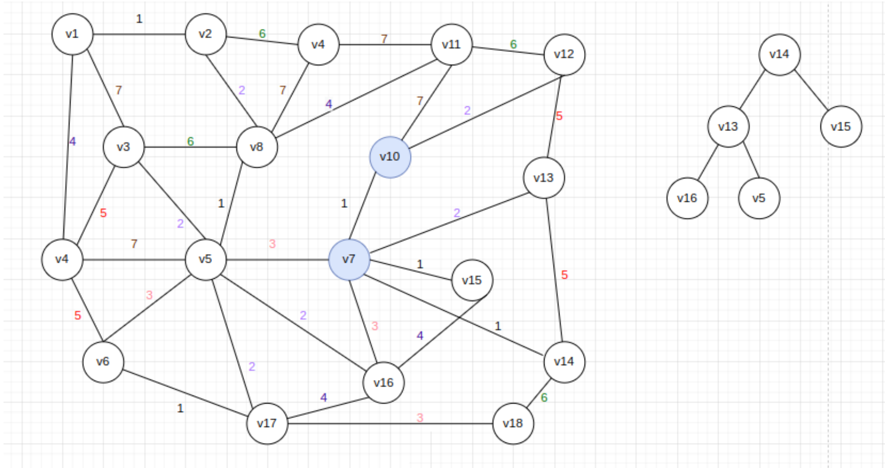
Ver	Pad	Cos	Vis	Ite
v_1	x	∞	F	F
v_2	x	∞	F	F
v_3	x	∞	F	F
v_4	x	∞	F	F
v_5	v_7	3	T	F
v_6	x	∞	F	F
v_7	x	∞	T	T
v_8	x	∞	F	F
v_9	x	∞	F	F
v_{10}	v_7	1	T	F
v_{11}	x	∞	F	F
v_{12}	x	∞	F	F
v_{13}	v_7	2	T	F
v_{14}	v_7	1	T	F
v_{15}	v_7	1	T	F
v_{16}	x	∞	F	F
v_{17}	x	∞	F	F
v_{18}	x	∞	F	F



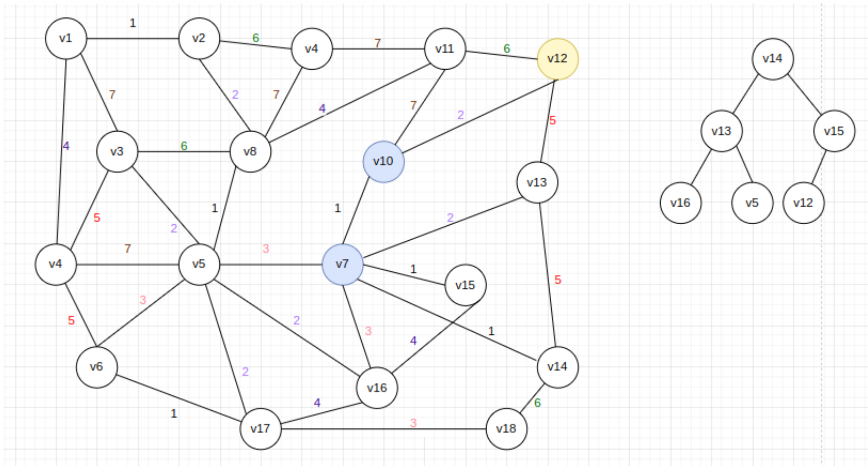
Ver	Pad	Cos	Vis	Ite
v_1	x	∞	F	F
v_2	x	∞	F	F
v_3	x	∞	F	F
v_4	x	∞	F	F
v_5	v_7	3	T	F
v_6	x	∞	F	F
v_7	x	∞	T	T
v_8	x	∞	F	F
v_9	x	∞	F	F
v_{10}	v_7	1	T	F
v_{11}	x	∞	F	F
v_{12}	x	∞	F	F
v_{13}	v_7	2	T	F
v_{14}	v_7	1	T	F
v_{15}	v_7	1	T	F
v_{16}	v_7	3	T	F
v_{17}	x	∞	F	F
v_{18}	x	∞	F	F



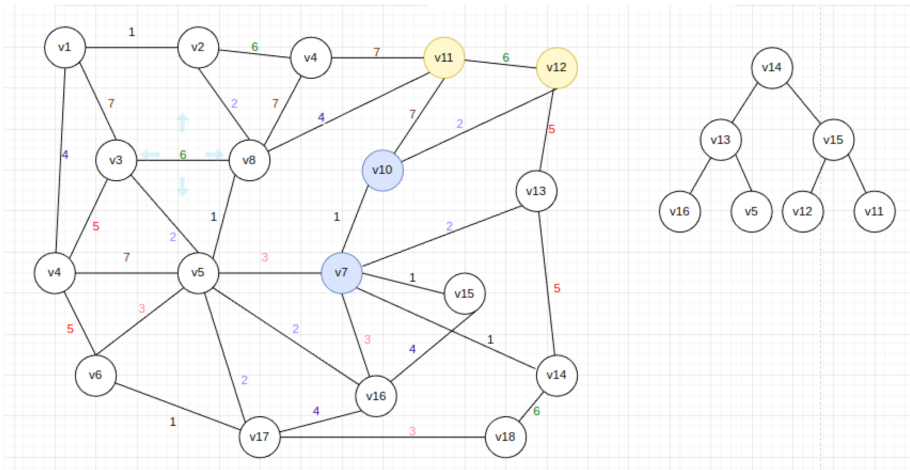
Ver	Pad	Cos	Vis	Ite
v_1	x	∞	F	F
v_2	x	∞	F	F
v_3	x	∞	F	F
v_4	x	∞	F	F
v_5	v_7	3	T	F
v_6	x	∞	F	F
v_7	x	∞	T	T
v_8	x	∞	F	F
v_9	x	∞	F	F
v_{10}	v_7	1	T	T
v_{11}	x	∞	F	F
v_{12}	x	∞	F	F
v_{13}	v_7	2	T	F
v_{14}	v_7	1	T	F
v_{15}	v_7	1	T	F
v_{16}	v_7	3	T	F
v_{17}	x	∞	F	F
v_{18}	x	∞	F	F



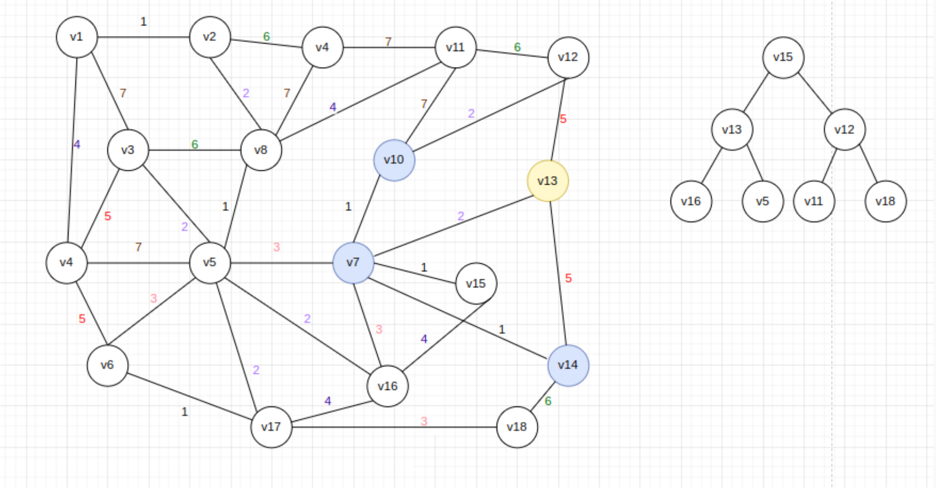
Ver	Pad	Cos	Vis	Ite
v_1	x	∞	F	F
v_2	x	∞	F	F
v_3	x	∞	F	F
v_4	x	∞	F	F
v_5	v_7	3	T	F
v_6	x	∞	F	F
v_7	x	∞	T	T
v_8	x	∞	F	F
v_9	x	∞	F	F
v_{10}	v_7	1	T	T
v_{11}	x	∞	F	F
v_{12}	v_{10}	2	T	F
v_{13}	v_7	2	T	F
v_{14}	v_7	1	T	F
v_{15}	v_7	1	T	F
v_{16}	v_7	3	T	F
v_{17}	x	∞	F	F
v_{18}	x	∞	F	F



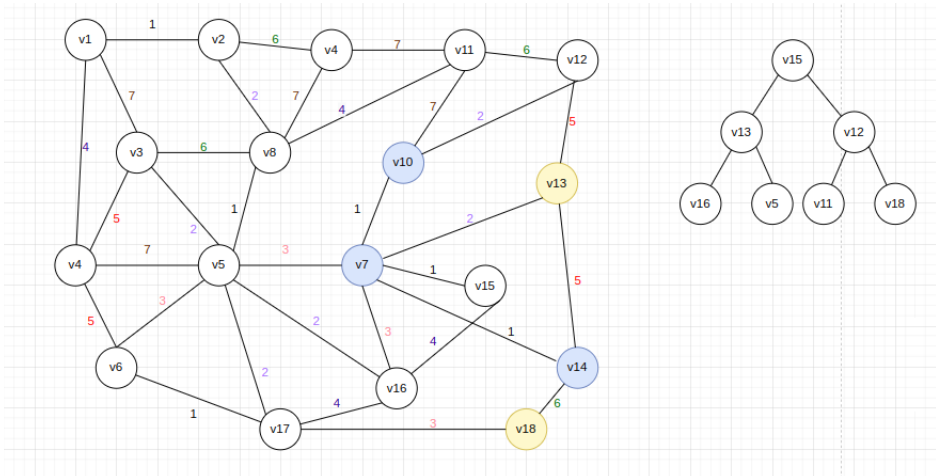
Ver	Pad	Cos	Vis	Ite
v_1	x	∞	F	F
v_2	x	∞	F	F
v_3	x	∞	F	F
v_4	x	∞	F	F
v_5	v_7	3	T	F
v_6	x	∞	F	F
v_7	x	∞	T	T
v_8	x	∞	F	F
v_9	x	∞	F	F
v_{10}	v_7	1	T	T
v_{11}	v_{10}	7	T	F
v_{12}	v_{10}	2	T	F
v_{13}	v_7	2	T	F
v_{14}	v_7	1	T	F
v_{15}	v_7	1	T	F
v_{16}	v_7	3	T	F
v_{17}	x	∞	F	F
v_{18}	x	∞	F	F



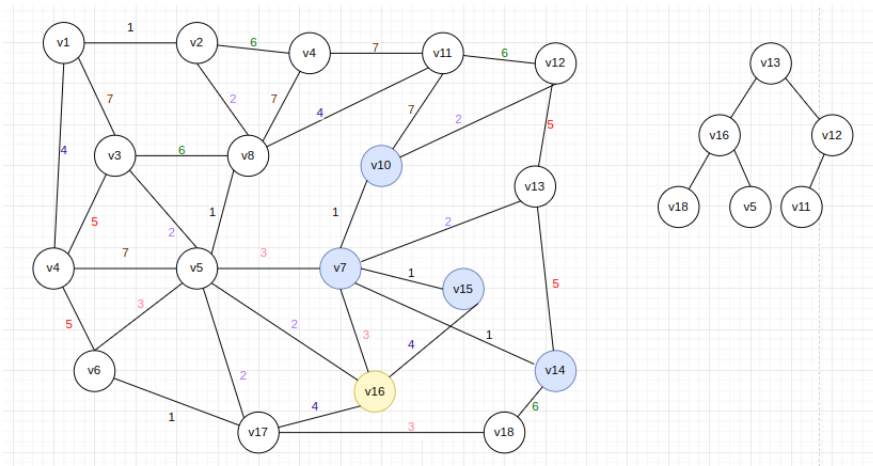
Ver	Pad	Cos	Vis	Ite
v_1	x	∞	F	F
v_2	x	∞	F	F
v_3	x	∞	F	F
v_4	x	∞	F	F
v_5	v_7	3	T	F
v_6	x	∞	F	F
v_7	x	∞	T	T
v_8	x	∞	F	F
v_9	x	∞	F	F
v_{10}	v_7	1	T	T
v_{11}	v_{10}	7	T	F
v_{12}	v_{10}	2	T	F
v_{13}	v_7	2	T	F
v_{14}	v_7	1	T	F
v_{15}	v_7	1	T	F
v_{16}	v_7	3	T	F
v_{17}	x	∞	F	F
v_{18}	x	∞	F	F



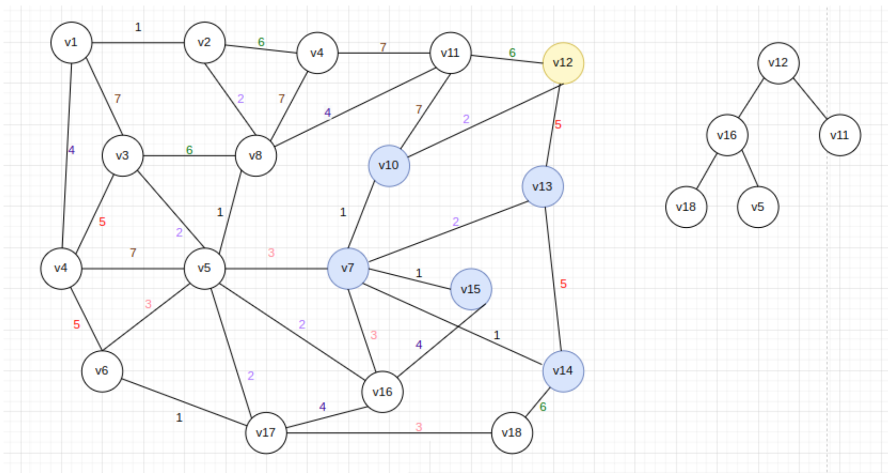
Ver	Pad	Cos	Vis	Ite
v_1	x	∞	F	F
v_2	x	∞	F	F
v_3	x	∞	F	F
v_4	x	∞	F	F
v_5	v_7	3	T	F
v_6	x	∞	F	F
v_7	x	∞	T	T
v_8	x	∞	F	F
v_9	x	∞	F	F
v_{10}	v_7	1	T	T
v_{11}	v_{10}	7	T	F
v_{12}	v_{10}	2	T	F
v_{13}	v_7	2	T	F
v_{14}	v_7	1	T	F
v_{15}	v_7	1	T	F
v_{16}	v_7	3	T	F
v_{17}	x	∞	F	F
v_{18}	v_{14}	6	T	F



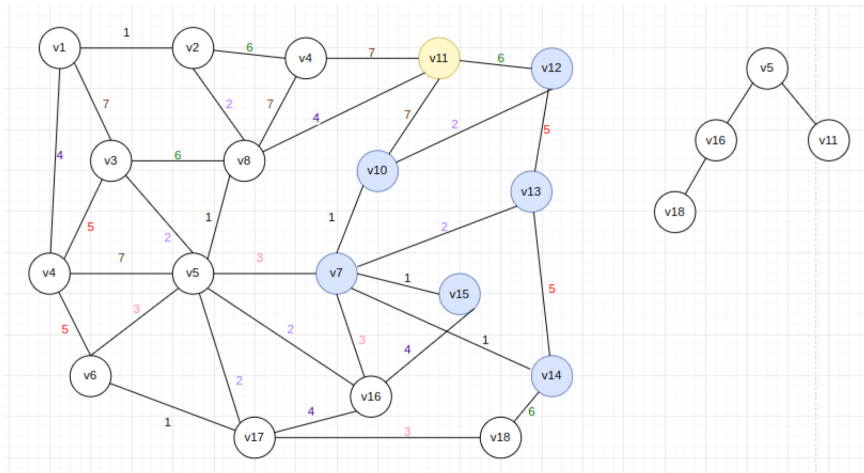
Ver	Pad	Cos	Vis	Ite
v_1	x	∞	F	F
v_2	x	∞	F	F
v_3	x	∞	F	F
v_4	x	∞	F	F
v_5	v_7	3	T	F
v_6	x	∞	F	F
v_7	x	∞	T	T
v_8	x	∞	F	F
v_9	x	∞	F	F
v_{10}	v_7	1	T	T
v_{11}	v_{10}	7	T	F
v_{12}	v_{10}	2	T	F
v_{13}	v_7	2	T	F
v_{14}	v_7	1	T	T
v_{15}	v_7	1	T	F
v_{16}	v_7	3	T	F
v_{17}	x	∞	F	F
v_{18}	v_{14}	6	T	F



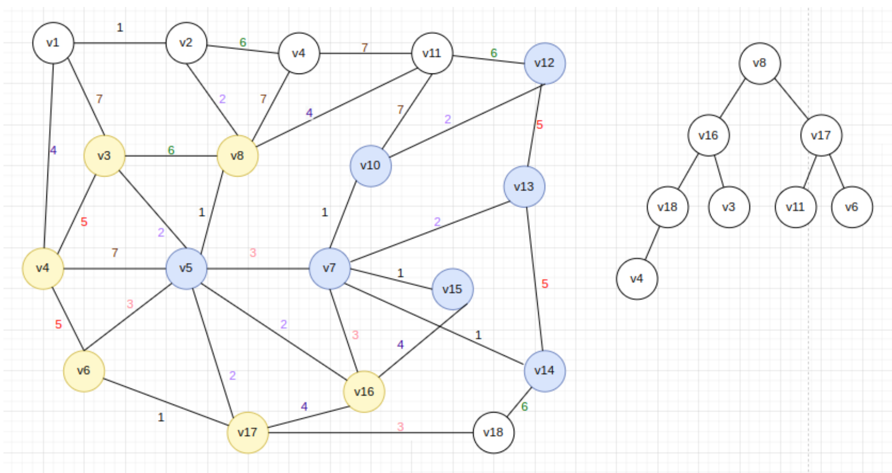
Ver	Pad	Cos	Vis	Ite
v_1	x	∞	F	F
v_2	x	∞	F	F
v_3	x	∞	F	F
v_4	x	∞	F	F
v_5	v_7	3	T	F
v_6	x	∞	F	F
v_7	x	∞	T	T
v_8	x	∞	F	F
v_9	x	∞	F	F
v_{10}	v_7	1	T	T
v_{11}	v_{10}	7	T	F
v_{12}	v_{10}	2	T	F
v_{13}	v_7	2	T	T
v_{14}	v_7	1	T	F
v_{15}	v_7	1	T	T
v_{16}	v_7	3	T	F
v_{17}	x	∞	F	F
v_{18}	v_{14}	6	T	F



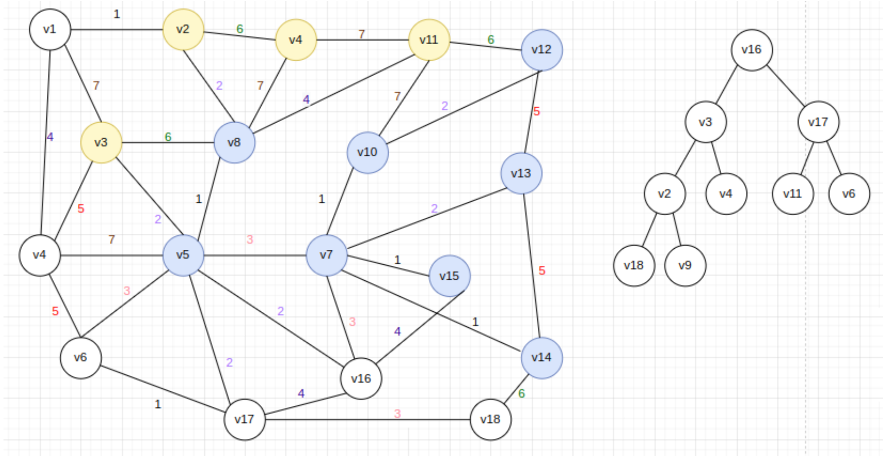
Ver	Pad	Cos	Vis	Ite
v_1	x	∞	F	F
v_2	x	∞	F	F
v_3	x	∞	F	F
v_4	x	∞	F	F
v_5	v_7	3	T	F
v_6	x	∞	F	F
v_7	x	∞	T	T
v_8	x	∞	F	F
v_9	x	∞	F	F
v_{10}	v_7	1	T	T
v_{11}	v_6	6	T	F
v_{12}	v_{10}	2	T	T
v_{13}	v_7	2	T	T
v_{14}	v_7	1	T	F
v_{15}	v_7	1	T	T
v_{16}	v_7	3	T	F
v_{17}	x	∞	F	F
v_{18}	v_{14}	6	T	F



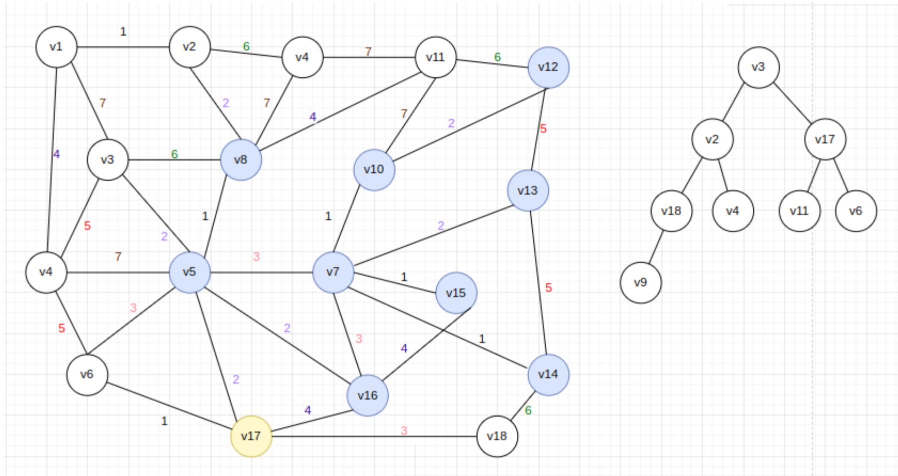
Ver	Pad	Cos	Vis	Ite
v_1	x	∞	F	F
v_2	x	∞	F	F
v_3	v_5	2	T	F
v_4	v_5	7	T	F
v_5	v_7	3	T	T
v_6	v_5	3	T	F
v_7	x	∞	T	T
v_8	v_5	1	T	F
v_9	x	∞	F	F
v_{10}	v_7	1	T	T
v_{11}	v_6	6	T	F
v_{12}	v_{10}	2	T	T
v_{13}	v_7	2	T	T
v_{14}	v_7	1	T	F
v_{15}	v_7	1	T	T
v_{16}	v_7	3	T	F
v_{17}	v_5	2	T	F
v_{18}	v_{14}	6	T	F



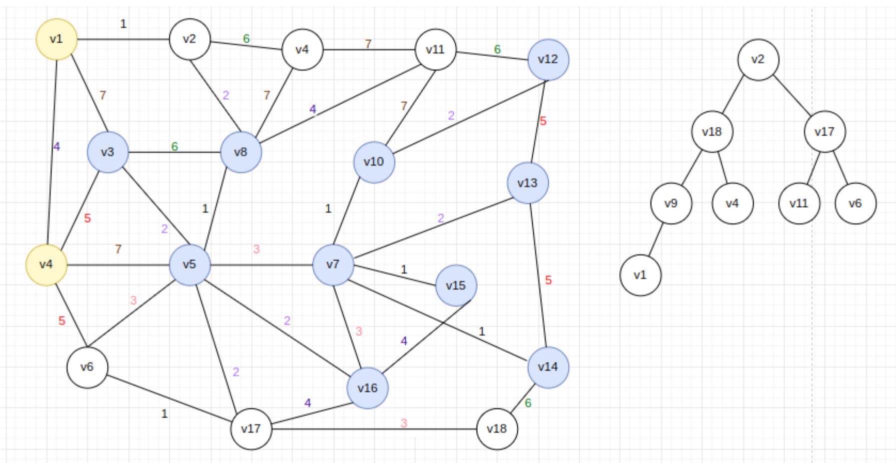
Ver	Pad	Cos	Vis	Ite
v_1	x	∞	F	F
v_2	v_8	2	T	F
v_3	v_5	2	T	F
v_4	v_5	7	T	F
v_5	v_7	3	T	T
v_6	v_5	3	T	F
v_7	x	∞	T	T
v_8	v_5	1	T	F
v_9	v_8	7	T	F
v_{10}	v_7	1	T	T
v_{11}	v_8	4	T	F
v_{12}	v_{10}	2	T	T
v_{13}	v_7	2	T	T
v_{14}	v_7	1	T	F
v_{15}	v_7	1	T	T
v_{16}	v_5	2	T	F
v_{17}	v_5	2	T	F
v_{18}	v_{14}	6	T	F



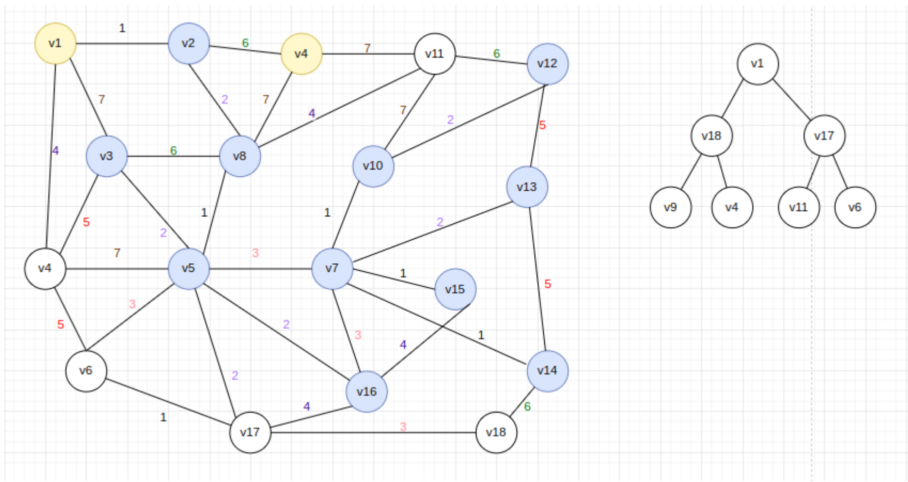
Ver	Pad	Cos	Vis	Ite
v_1	x	∞	F	F
v_2	v_8	2	T	F
v_3	v_5	2	T	F
v_4	v_5	7	T	F
v_5	v_7	3	T	T
v_6	v_5	3	T	F
v_7	x	∞	T	T
v_8	v_5	1	T	F
v_9	v_8	7	T	F
v_{10}	v_7	1	T	T
v_{11}	v_8	4	T	F
v_{12}	v_{10}	2	T	T
v_{13}	v_7	2	T	T
v_{14}	v_7	1	T	F
v_{15}	v_7	1	T	T
v_{16}	v_5	2	T	T
v_{17}	v_5	2	T	F
v_{18}	v_{14}	6	T	F



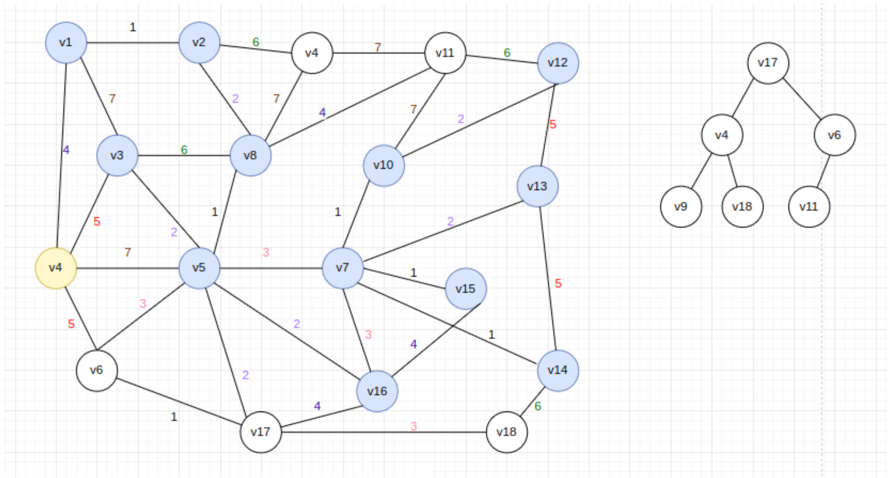
Ver	Pad	Cos	Vis	Ite
v_1	v_3	7	T	F
v_2	v_8	2	T	F
v_3	v_5	2	T	T
v_4	v_5	7	T	F
v_5	v_7	3	T	T
v_6	v_5	3	T	F
v_7	x	∞	T	T
v_8	v_5	1	T	F
v_9	v_8	7	T	F
v_{10}	v_7	1	T	T
v_{11}	v_8	4	T	F
v_{12}	v_{10}	2	T	T
v_{13}	v_7	2	T	T
v_{14}	v_7	1	T	F
v_{15}	v_7	1	T	T
v_{16}	v_5	2	T	T
v_{17}	v_5	2	T	F
v_{18}	v_{14}	6	T	F



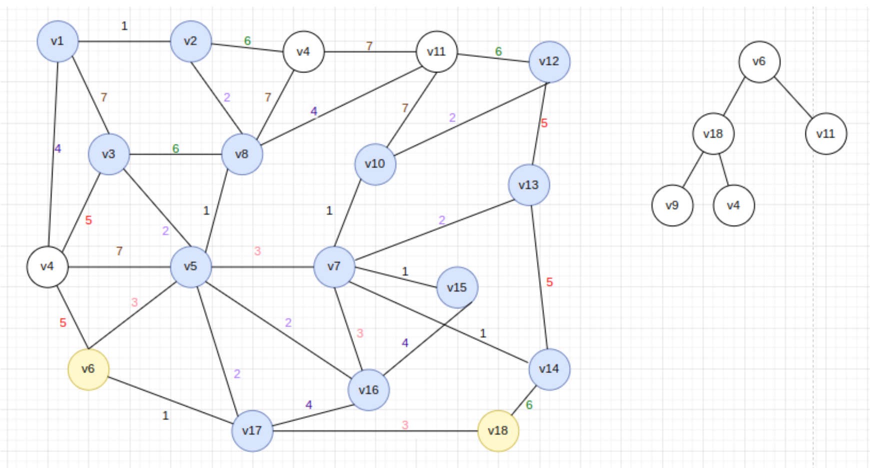
Ver	Pad	Cos	Vis	Ite
v_1	v_2	1	T	F
v_2	v_8	2	T	T
v_3	v_5	2	T	T
v_4	v_5	7	T	F
v_5	v_7	3	T	T
v_6	v_5	3	T	F
v_7	x	∞	T	T
v_8	v_5	1	T	F
v_9	v_2	6	T	F
v_{10}	v_7	1	T	T
v_{11}	v_8	4	T	F
v_{12}	v_{10}	2	T	T
v_{13}	v_7	2	T	T
v_{14}	v_7	1	T	F
v_{15}	v_7	1	T	T
v_{16}	v_5	2	T	T
v_{17}	v_5	2	T	F
v_{18}	v_{14}	6	T	F



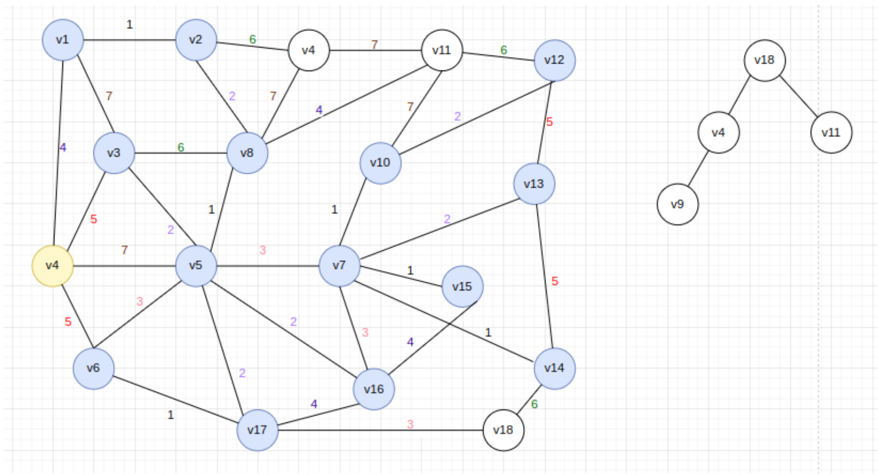
Ver	Pad	Cos	Vis	Ite
v_1	v_2	1	T	T
v_2	v_8	2	T	T
v_3	v_5	2	T	T
v_4	v_1	4	T	F
v_5	v_7	3	T	T
v_6	v_5	3	T	F
v_7	x	∞	T	T
v_8	v_5	1	T	F
v_9	v_2	6	T	F
v_{10}	v_7	1	T	T
v_{11}	v_8	4	T	F
v_{12}	v_{10}	2	T	T
v_{13}	v_7	2	T	T
v_{14}	v_7	1	T	F
v_{15}	v_7	1	T	T
v_{16}	v_5	2	T	T
v_{17}	v_5	2	T	F
v_{18}	v_{14}	6	T	F



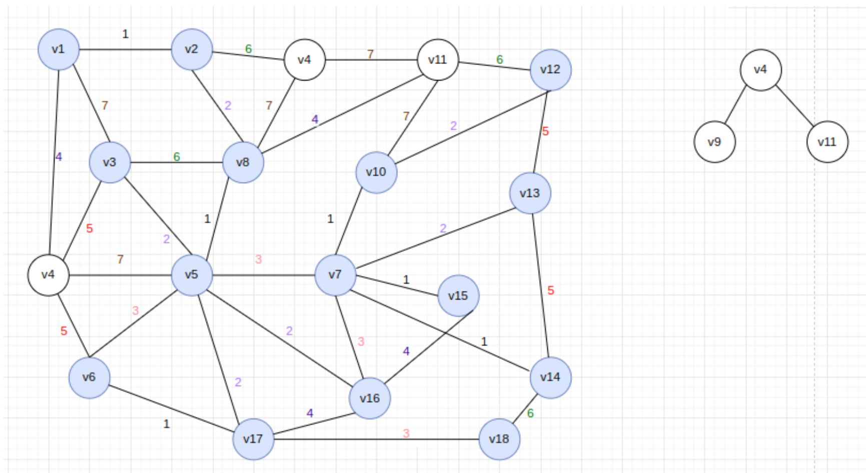
Ver	Pad	Cos	Vis	Ite
v_1	v_2	1	T	T
v_2	v_8	2	T	T
v_3	v_5	2	T	T
v_4	v_1	4	T	F
v_5	v_7	3	T	T
v_6	v_{17}	1	T	F
v_7	x	∞	T	T
v_8	v_5	1	T	T
v_9	v_2	6	T	F
v_{10}	v_7	1	T	T
v_{11}	v_8	4	T	F
v_{12}	v_{10}	2	T	T
v_{13}	v_7	2	T	T
v_{14}	v_7	1	T	T
v_{15}	v_7	1	T	T
v_{16}	v_5	2	T	T
v_{17}	v_5	2	T	T
v_{18}	v_{17}	3	T	F



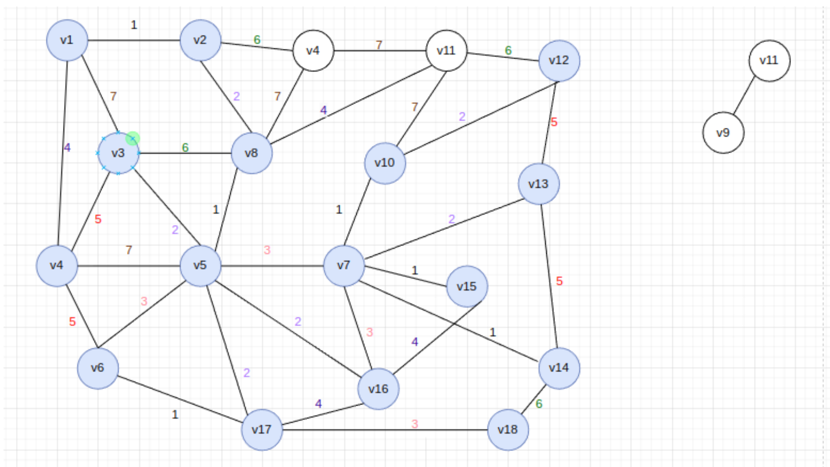
Ver	Pad	Cos	Vis	Ite
v_1	v_2	1	T	T
v_2	v_8	2	T	T
v_3	v_5	2	T	T
v_4	v_1	4	T	F
v_5	v_7	3	T	T
v_6	v_{17}	1	T	T
v_7	x	∞	T	T
v_8	v_5	1	T	T
v_9	v_2	6	T	F
v_{10}	v_7	1	T	T
v_{11}	v_8	4	T	F
v_{12}	v_{10}	2	T	T
v_{13}	v_7	2	T	T
v_{14}	v_7	1	T	T
v_{15}	v_7	1	T	T
v_{16}	v_5	2	T	T
v_{17}	v_5	2	T	T
v_{18}	v_{17}	3	T	F



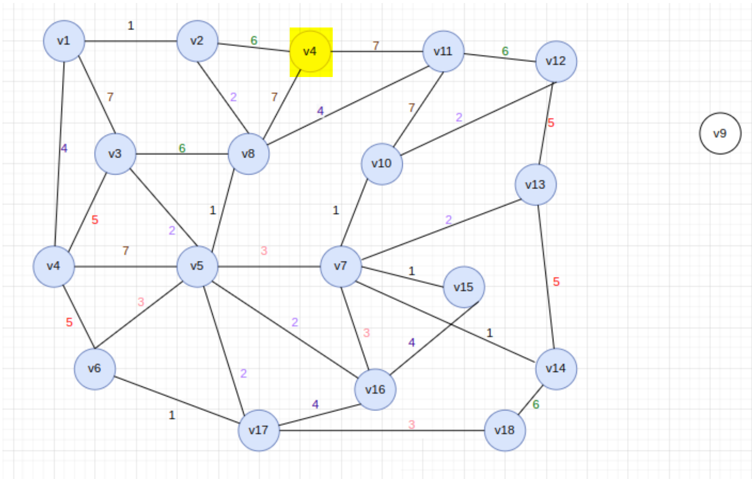
Ver	Pad	Cos	Vis	Ite
v_1	v_2	1	T	T
v_2	v_8	2	T	T
v_3	v_5	2	T	T
v_4	v_1	4	T	F
v_5	v_7	3	T	T
v_6	v_{17}	1	T	T
v_7	x	∞	T	T
v_8	v_5	1	T	T
v_9	v_2	6	T	F
v_{10}	v_7	1	T	T
v_{11}	v_8	4	T	F
v_{12}	v_{10}	2	T	T
v_{13}	v_7	2	T	T
v_{14}	v_7	1	T	T
v_{15}	v_7	1	T	T
v_{16}	v_5	2	T	T
v_{17}	v_5	2	T	T
v_{18}	v_{17}	3	T	T



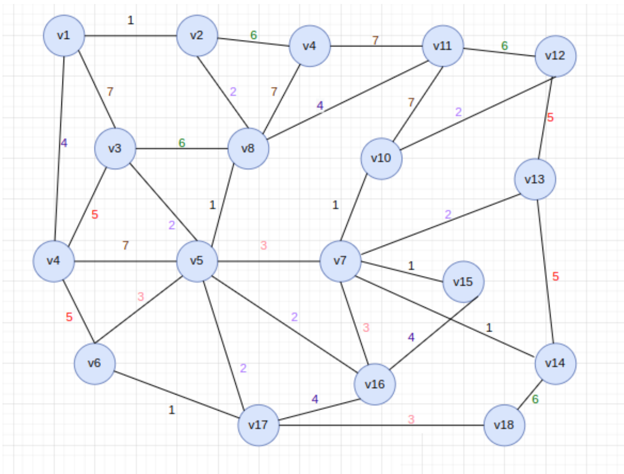
Ver	Pad	Cos	Vis	Ite
v_1	v_2	1	T	T
v_2	v_8	2	T	T
v_3	v_5	2	T	T
v_4	v_1	4	T	T
v_5	v_7	3	T	T
v_6	v_{17}	1	T	T
v_7	x	∞	T	T
v_8	v_5	1	T	T
v_9	v_2	6	T	F
v_{10}	v_7	1	T	T
v_{11}	v_8	4	T	F
v_{12}	v_{10}	2	T	T
v_{13}	v_7	2	T	T
v_{14}	v_7	1	T	T
v_{15}	v_7	1	T	T
v_{16}	v_5	2	T	T
v_{17}	v_5	2	T	T
v_{18}	v_{17}	3	T	T



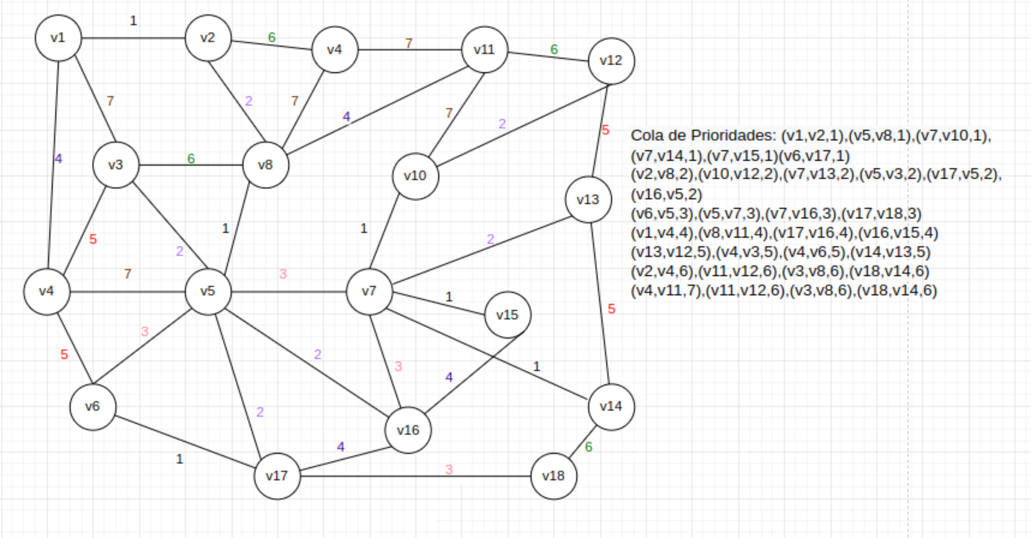
Ver	Pad	Cos	Vis	Ite
v_1	v_2	1	T	T
v_2	v_8	2	T	T
v_3	v_5	2	T	T
v_4	v_1	4	T	T
v_5	v_7	3	T	T
v_6	v_{17}	1	T	T
v_7	x	∞	T	T
v_8	v_5	1	T	T
v_9	v_2	6	T	F
v_{10}	v_7	1	T	T
v_{11}	v_8	4	T	T
v_{12}	v_{10}	2	T	T
v_{13}	v_7	2	T	T
v_{14}	v_7	1	T	T
v_{15}	v_7	1	T	T
v_{16}	v_5	2	T	T
v_{17}	v_5	2	T	T
v_{18}	v_{17}	3	T	T

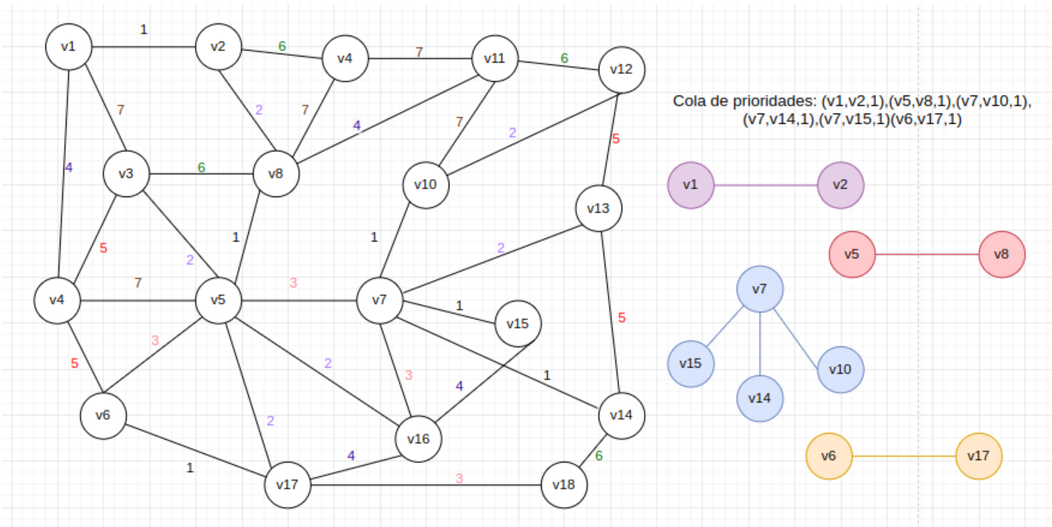


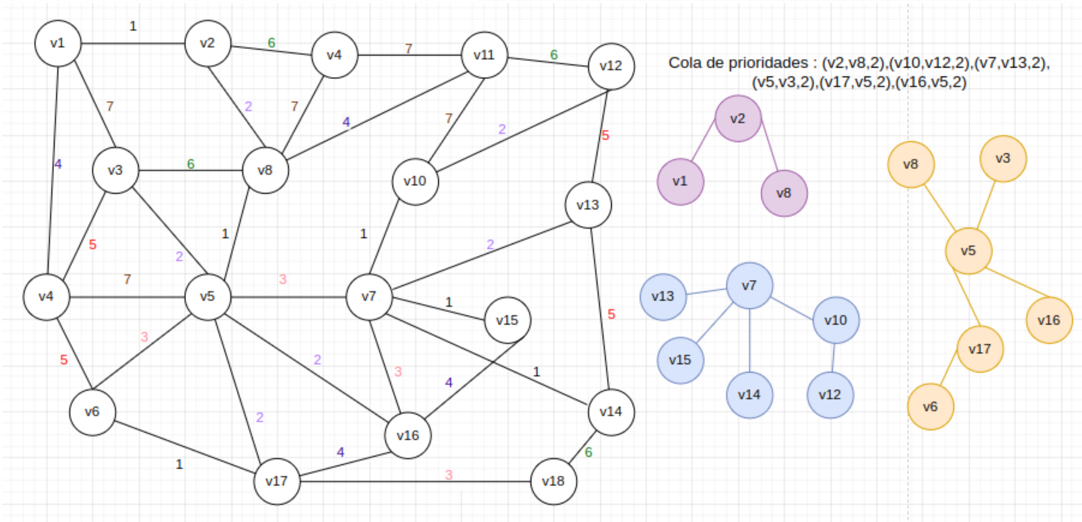
Ver	Pad	Cos	Vis	Ite
v_1	v_2	1	T	T
v_2	v_8	2	T	T
v_3	v_5	2	T	T
v_4	v_1	4	T	T
v_5	v_7	3	T	T
v_6	v_{17}	1	T	T
v_7	x	∞	T	T
v_8	v_5	1	T	T
v_9	v_2	6	T	T
v_{10}	v_7	1	T	T
v_{11}	v_8	4	T	T
v_{12}	v_{10}	2	T	T
v_{13}	v_7	2	T	T
v_{14}	v_7	1	T	T
v_{15}	v_7	1	T	T
v_{16}	v_5	2	T	T
v_{17}	v_5	2	T	T
v_{18}	v_{17}	3	T	T

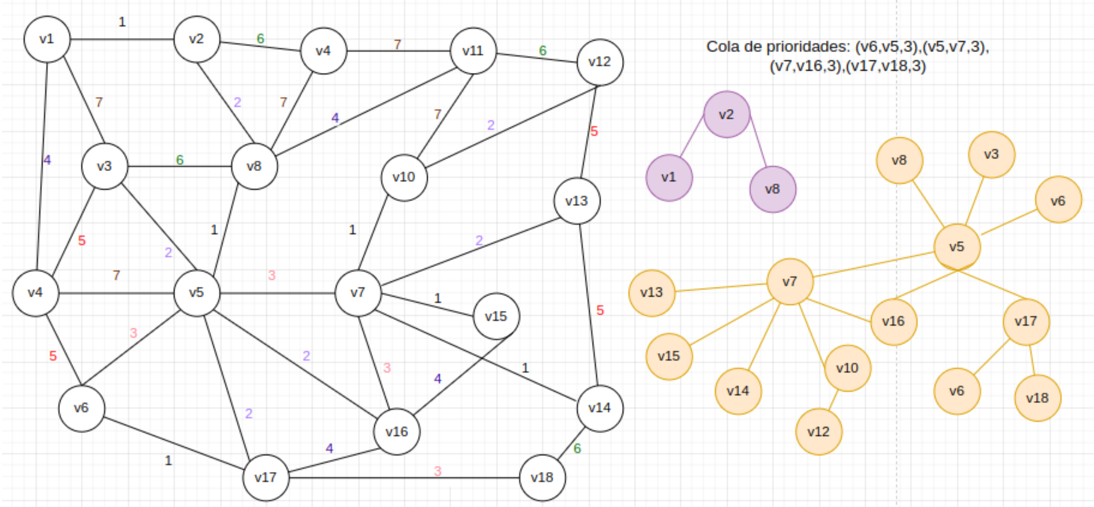


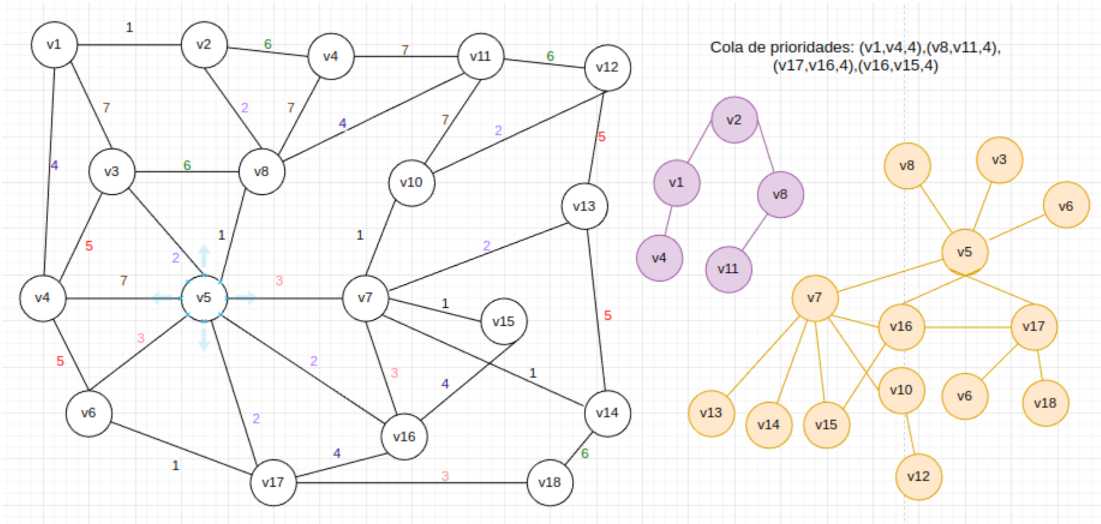
b) Kruskal usando Conjuntos Ajenos con unión por tamaño.

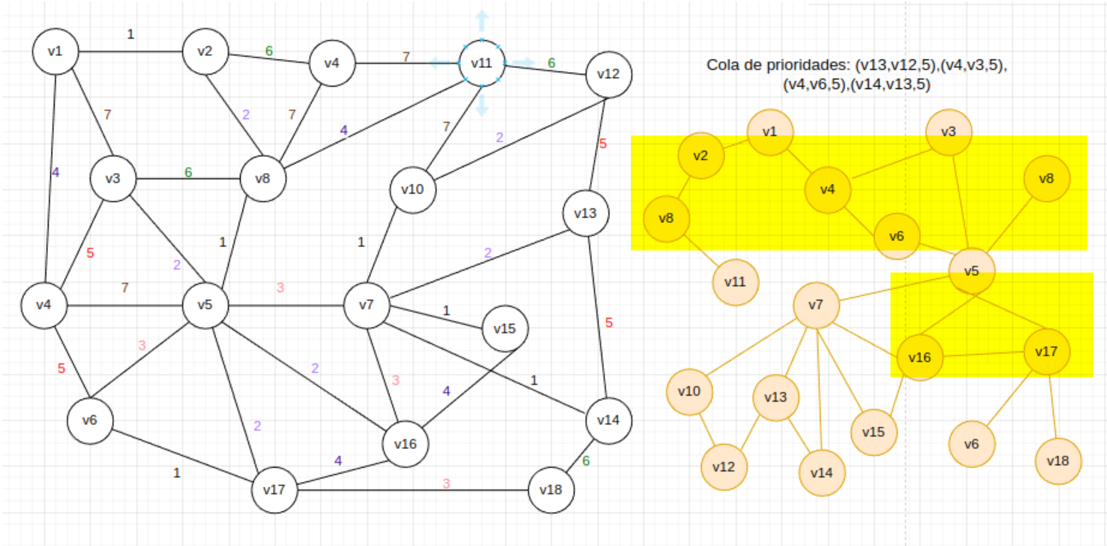


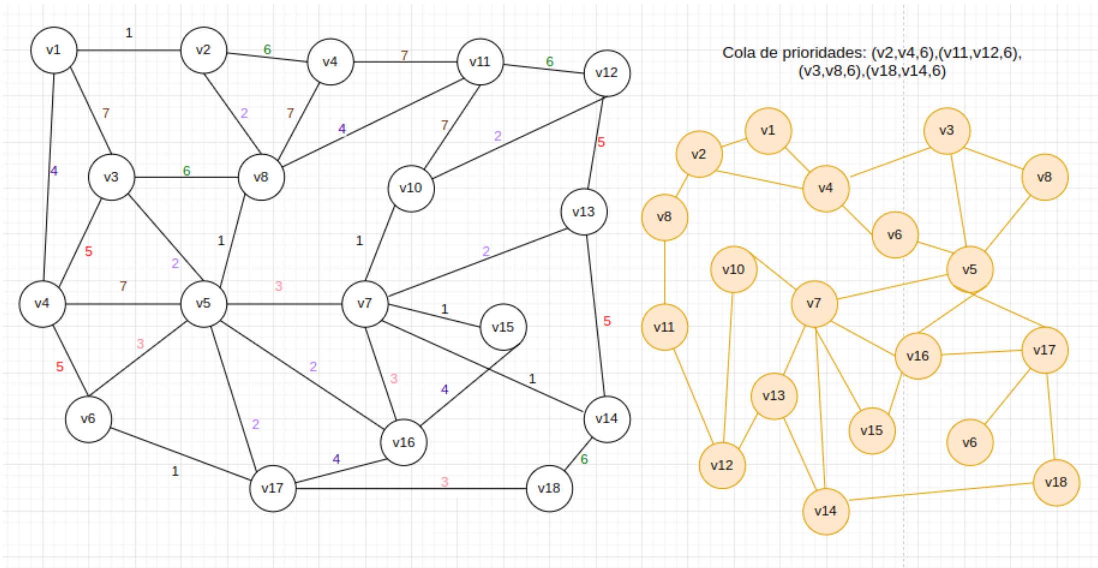


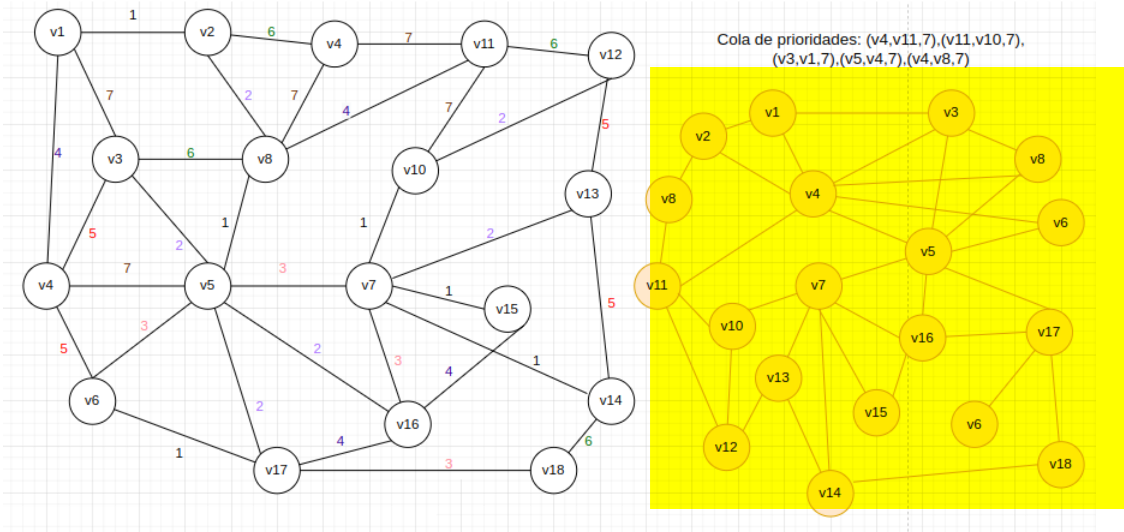



Cola de prioridades: (v6,v5,3),(v5,v7,3),
(v7,v16,3),(v17,v18,3)


Cola de prioridades: (v1,v4,4),(v8,v11,4),
(v17,v16,4),(v16,v15,4)


Cola de prioridades: (v13,v12,5),(v4,v3,5),
(v4,v6,5),(v14,v13,5)





c) Dijkstra usando Heap Binarios o Colas Binomiales.
Para la gráfica G = (V, A) dada, seleccionar un vértice como s, fuente, y dar dirección a las aristas para aplicar Dijsktra.
Respuesta

2. Sea G = (V, A) una gráfica conexa con pesos positivos sobre las aristas. Supongamos que el costo de un árbol generador se define como el producto de los costos en las aristas
   a) Diseñar un algoritmo que determine el árbol generador de peso máximo, bajo esta regla.
   b) Calcular el desempeño computacional del algoritmo propuesto, indicando las estructuras de datos usadas para lograr tal desempeño.
   Respuesta
   a) Ya que las aristas tienen pesos positivos podemos usar el inverso aditivo de cada número, es decir invertir el orden:

   al usar **KRUSKAL** se genera un árbol con peso mínimo, que al volver a sustituir de nuevo por el inverso aditivo genera un árbol de peso mínimo.
   algoritmo árbol generador (G)
   Precondiciones:
   -G=(V,A) pesos de las aristas positivas
   1. VacA, si su peso es p, su nuevo peso es -p
   2. Ampliar KRUSKAL (G)
   3. Se repite el paso 1

   b) El paso 1, se ejecutan en $O(n)$. mientras que **KRUSKAL**, supongamos que usamos heap binario como cola de prioridades y conjuntos ajenos, la complejidad seria de $O(nlogm)$ con $|V| = n, |A| = m$

3. **[Opcional]** Sea a una arista de peso mínimo de una gráfica G = (V, A) con pesos en las aristas. Modificar tanto el Algoritmo Prim como el Kruskal para que la arista a siempre aparezca en el árbol generador de peso mínimo.
   Respuesta
   Empezamos con **Prim**, entonces sean $b, c \in V$ tal que, b y c son vértices adyacentes de a, además de que queremos a siempre aparezca. Lo que debemos de hacer es que la primera inserción al +arbol T sean b y c (en vez de elegir un vértice arbitrario)
   1.Dar un valor inicial al conjunto T.
   Sean b, c, e, v, tal que b, c son adyacentes con a.
   T $\leftarrow$ $\{b, c\}$
   2.Sea e una arista de peso mínimo tal que un vértice de T con un vértice que no esté en T.
   T$\leftarrow$ T $\cup\{e\}$
   3. Revisar si el árbol generador ya ha sido completado.
   Si $|E(T) = n - T|$ entonces $returnE(T)$, en otro caso regresamos al paso

   Ahora con **Kruskal**, es un caso análogo, como buscamos que a siempre aparezca, basta con agregar desde el inicio a a al conjunto S
   1. Dar un valor al conjunto S. (insertando a)
   $S \xleftarrow{a}$
   2. Actualizar S
   Sea e una arista de peso mínimo tal que:
   $e \notin S$, tenemos que $S \cup \{e\}$ es cíclica
   Si existe tal e entonces: $S \leftarrow$ $S \cup \{e\}$
   3.Revisar si el árbol generador ya ha sido completado.
   Si $|S| = n - 1$ entonces $return$ S
   En otro caso regresamos al paso 2.