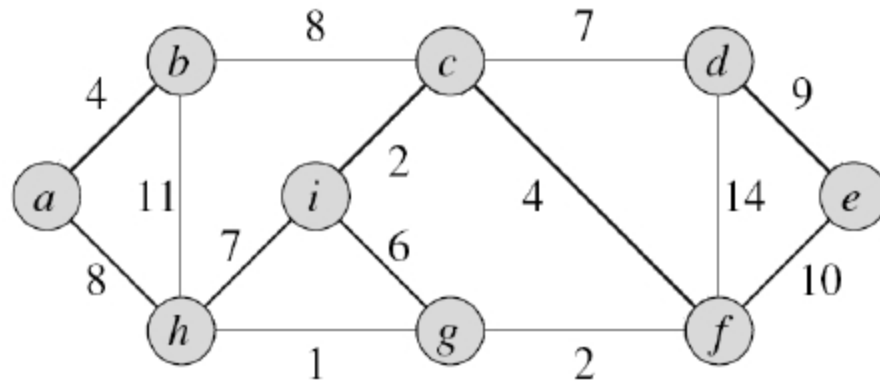


Algoritmo de PRIM

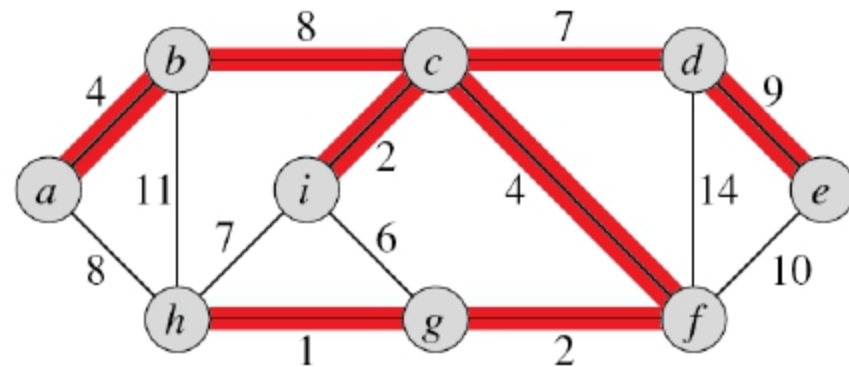
Relembrando aula passada... Kruskal

- Exemplo:

$$G = (V, A)$$



$$G' = (V, X)$$



Algoritmo de Prim

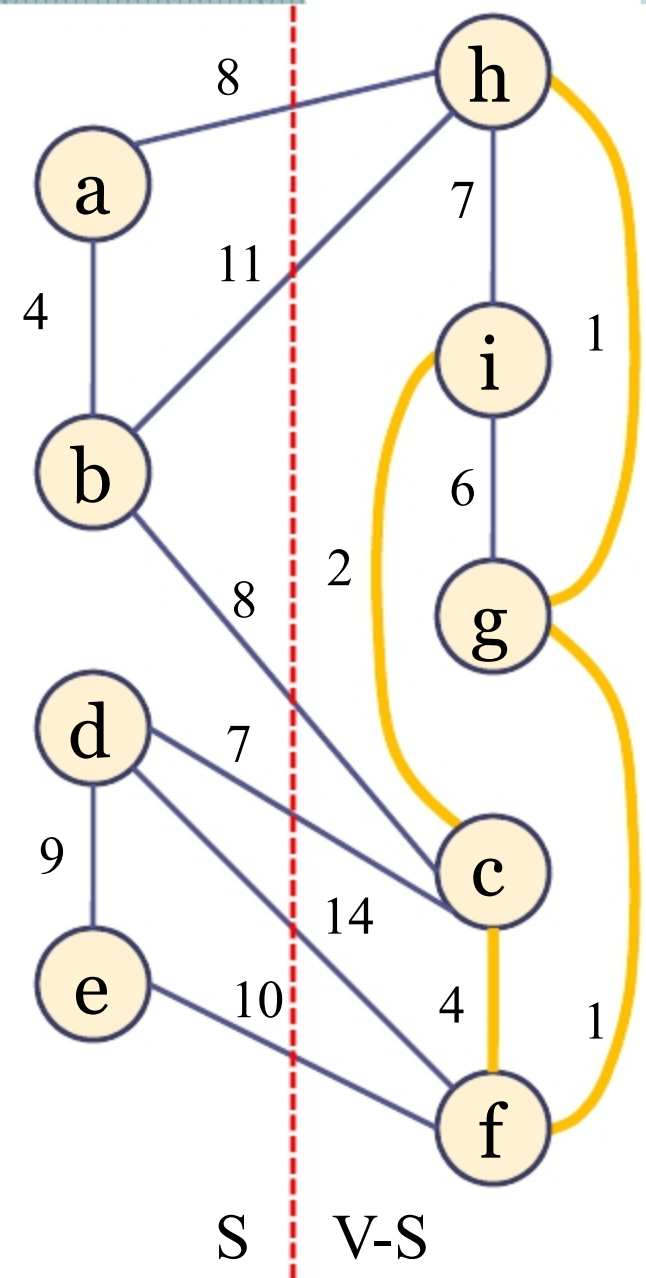
- Assim como o algoritmo de *Kruskal*, o algoritmo de *Prim* também é uma especialização do algoritmo genérico visto em sala;
 - Onde o ponto chave é a localização de uma aresta segura que cruza o corte para ingressar no conjunto X de arestas da AGM;

Algoritmo de Prim

- Ao contrário do *algoritmo de Kruskal*, *no algoritmo de Prim* o corte no grafo pode ser facilmente visualizado;
- e implementado...

$$S = \{a, b, d, e\}$$

$$V - S = \{h, i, c, g, f\}$$



Algoritmo de Prim

- Diferente do *algoritmo de Kruskal*, o *algoritmo de Prim* forma uma *única árvore* ao longo da construção da AGM;
- Idéia geral:
 - O algoritmo começa de um vértice r qualquer e expande a árvore até alcançar todos os vértices de V ;
 - Durante o processo, X é o tempo todo uma árvore; Nunca uma floresta com mais de uma árvore como pode acontecer no Algoritmo de Kruskal.

Algoritmo de Prim

- O *algoritmo de Prim* é **guloso** porque a árvore é expandida a cada etapa com uma aresta que **adiciona o menor peso possível na árvore**;
- Relembrando, o algoritmo de Kruskal é guloso porque adiciona a aresta de menor peso na floresta que não fecha ciclo em alguma de suas árvores;

Algoritmo de Prim

- O ponto chave para a obtenção de um algoritmo simples e eficiente **é tornar fácil a seleção de uma nova aresta** a ser adicionada à árvore formada pelas arestas em X;
 - Ou seja, das arestas que cruzam o corte, qual é a aresta leve?
Mas antes disso, quem é o corte???

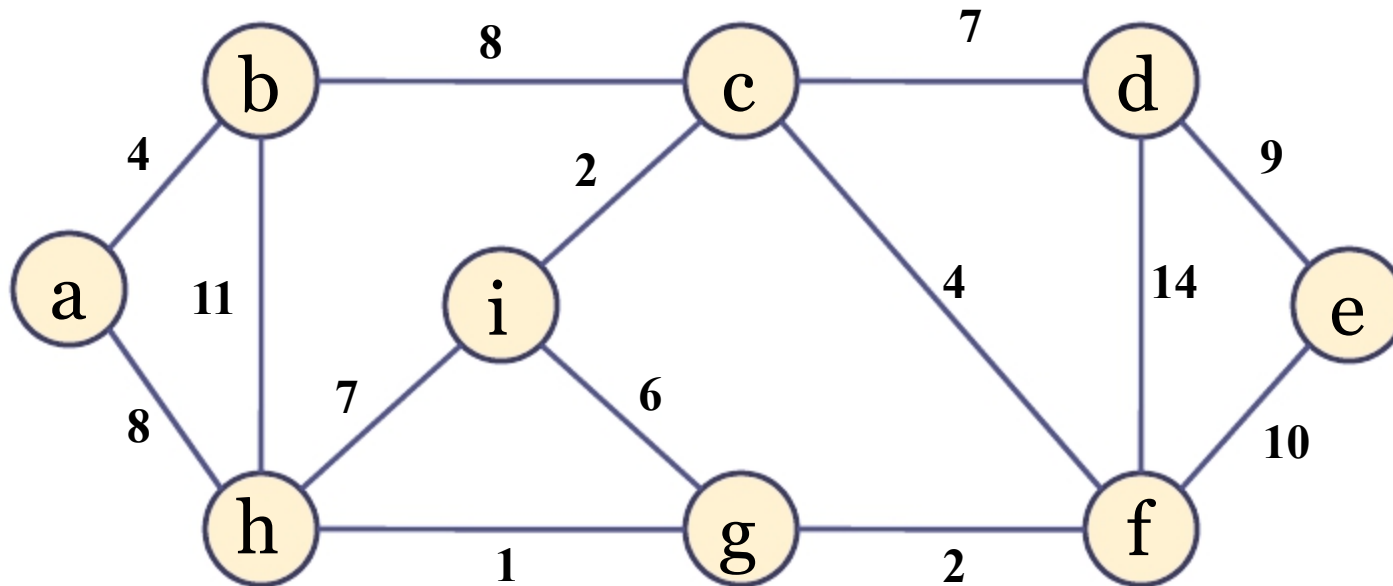
Algoritmo de Prim

- Estruturas auxiliares

- Q : Vértices que ainda não fazem parte da AGM parcial (ainda não fazem parte do conjunto X);
- $chave[u]$: peso da aresta mais leve do vértice u que a conecta à AGM parcialmente construída;
- $\pi[u]$: vértice pai do vértice u ;

Algoritmo de Prim

- Vamos acompanhar o algoritmo pelo seguinte grafo...



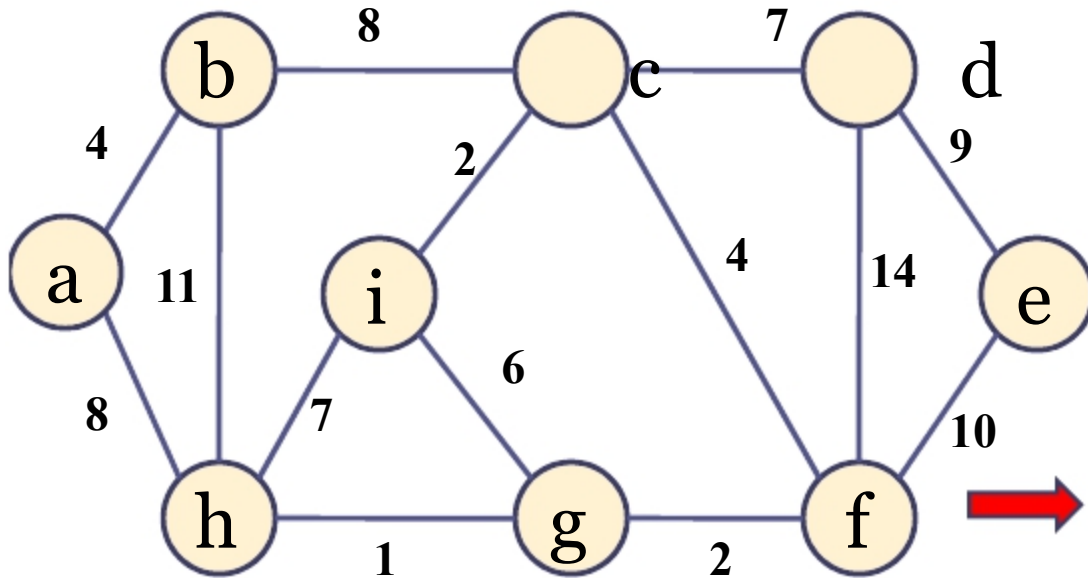
Algoritmo de Prim

$$\text{AGM_Prim}(G(V,A), w, r)$$
$$X \leftarrow \{ \}$$

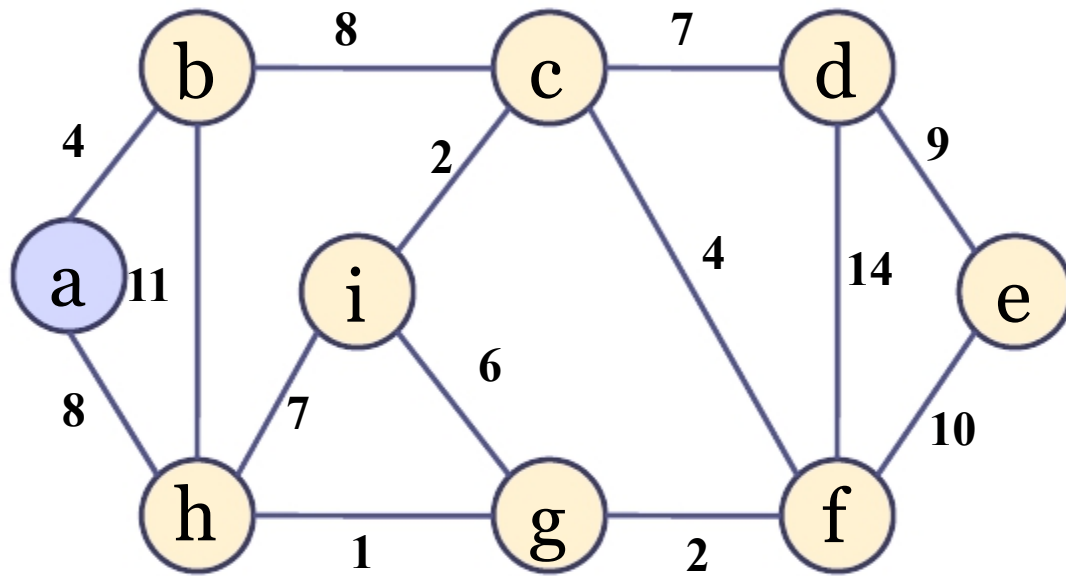
para cada vértice $u \in V$ faça

$$chave[u] \leftarrow \infty$$
$$\pi[u] \leftarrow NULL$$

fim para

$$chave[r] \leftarrow 0$$
$$Q \leftarrow V$$
[illegible]

Algoritmo de Prim



enquanto $Q \neq \{\}$



$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \text{ e } w(u, v) < \text{chave}[v]$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

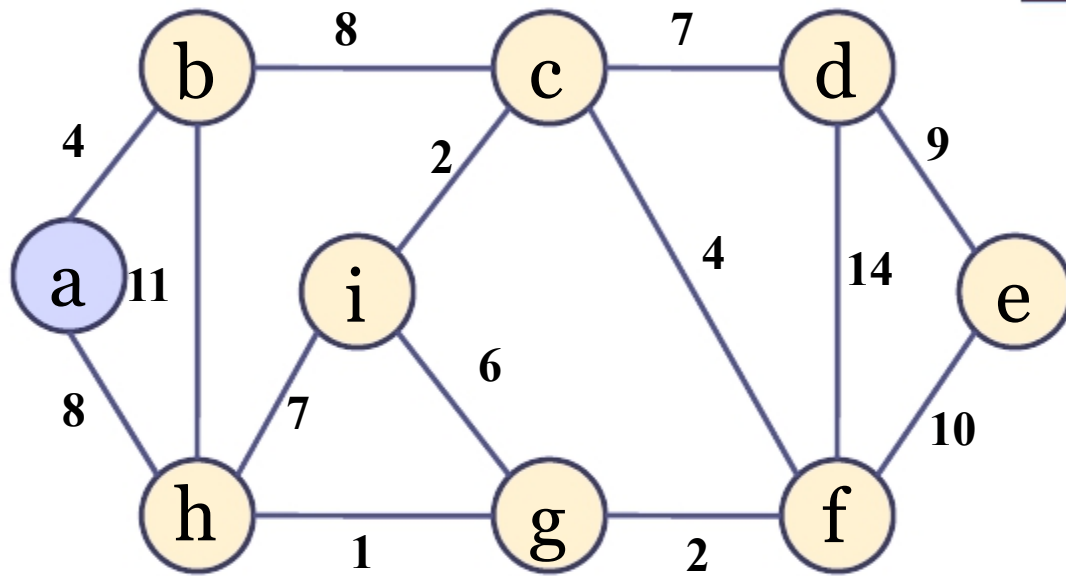
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	∞	∞	∞	∞	∞	∞	∞	∞
π	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
Q	---	X	X	X	X	X	X	X	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

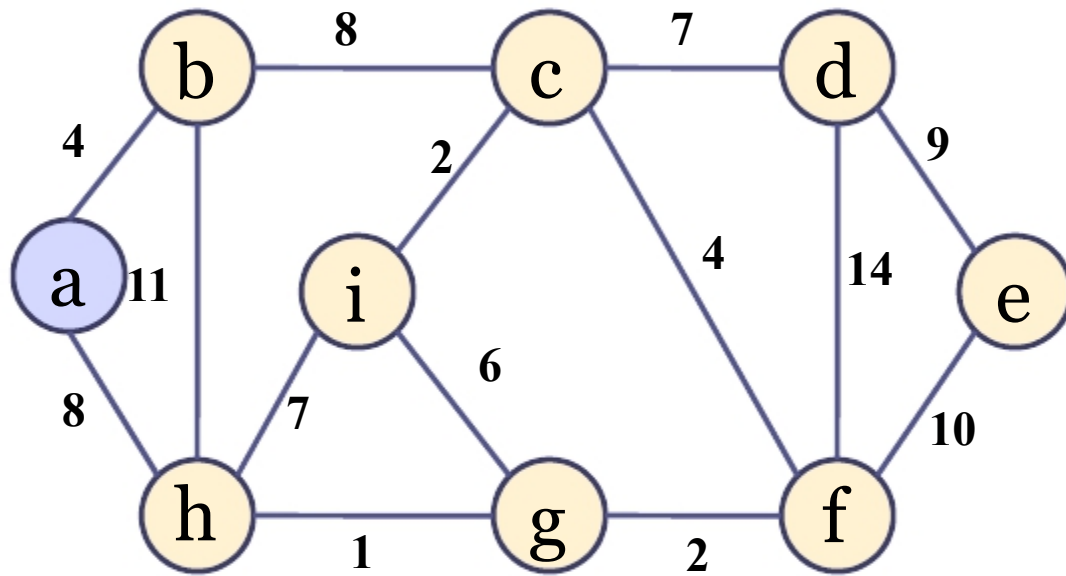
$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

→ para cada $v \in \text{Adj}[u]$ faça
 se $(v \in Q) \vee (w(u, v) < \text{chave}[v])$
 $\text{chave}[v] \leftarrow w(u, v)$
 $\pi[v] \leftarrow u$
 fimse
 fim para
 fim enquanto
 retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	∞	∞	∞	∞	∞	∞	∞	∞
π	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
Q	---	X	X	X	X	X	X	X	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \vee (w(u, v) < \text{chave}[v])$

→ $\text{chave}[v] \leftarrow w(u, v)$

→ $\pi[v] \leftarrow u$

fimse

fim para

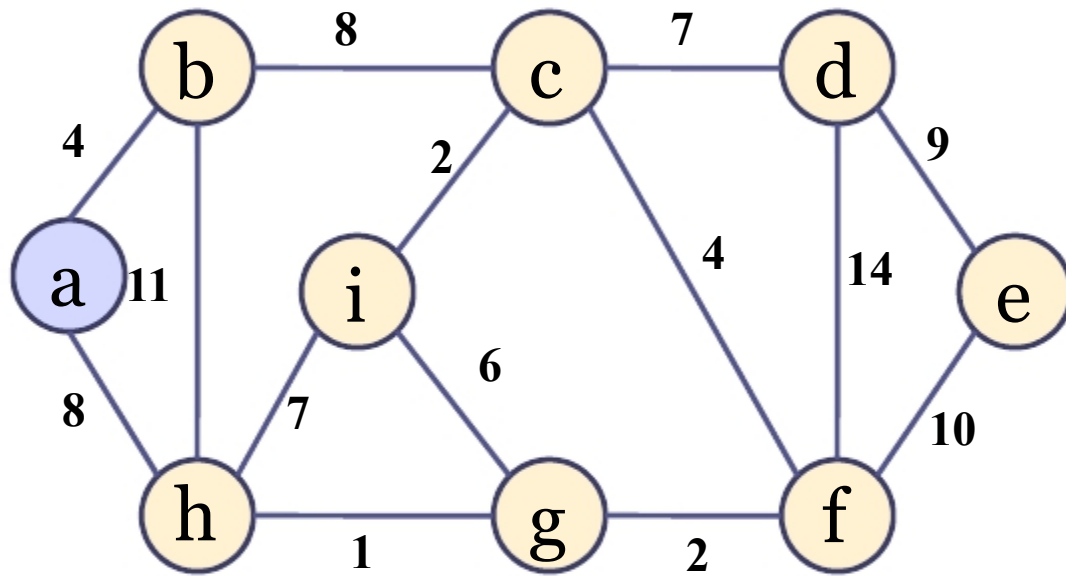
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	∞	∞	∞	∞	∞	8	∞
π	NULL	a	NULL	NULL	NULL	NULL	NULL	a	NULL
Q	---	X	X	X	X	X	X	X	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \text{ e } (w(u, v) < \text{chave}[v])$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

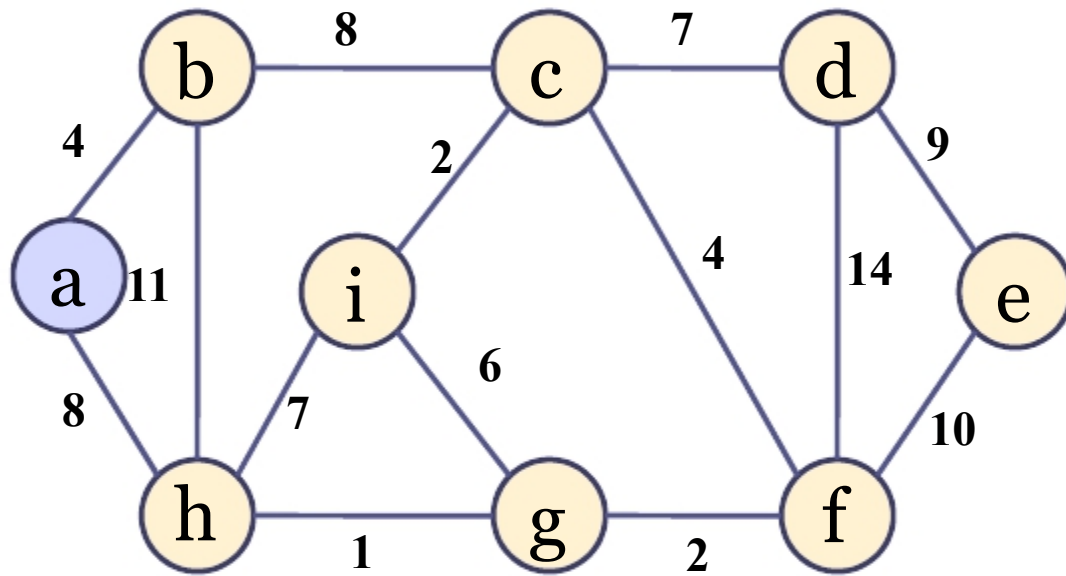
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	∞	∞	∞	∞	∞	8	∞
π	NULL	a	NULL	NULL	NULL	NULL	NULL	a	NULL
Q	---	X	X	X	X	X	X	X	X

Algoritmo de Prim



enquanto $Q \neq \{\}$



$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\}$ //seu $\neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \text{ e } (w(u, v) < \text{chave}[v])$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

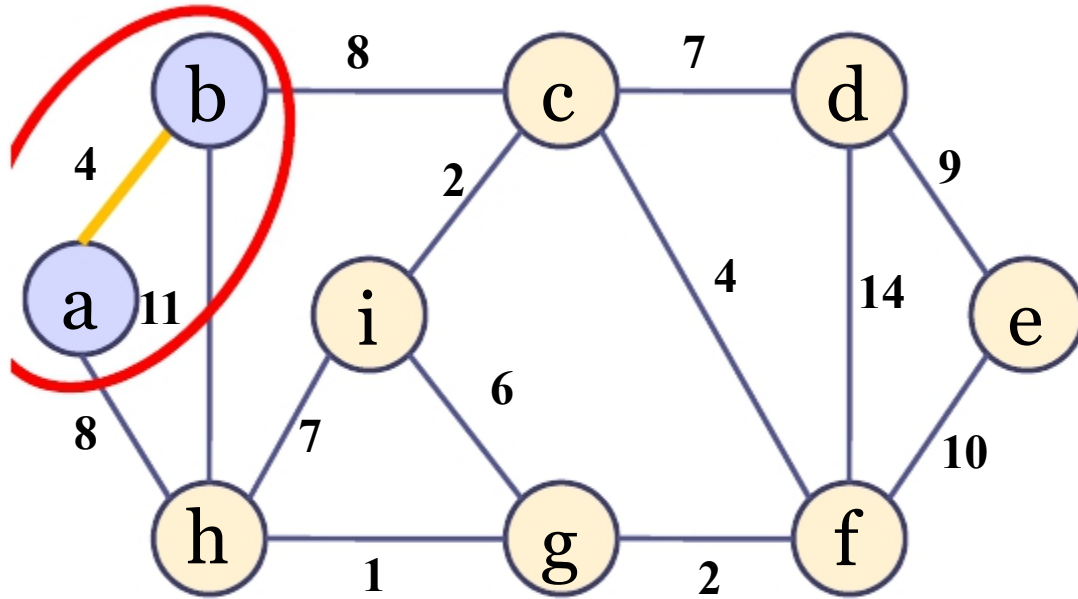
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	∞	∞	∞	∞	∞	8	∞
π	NULL	a	NULL	NULL	NULL	NULL	NULL	a	NULL
Q	---	---	X	X	X	X	X	X	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

→ $X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \wedge (w(u, v) < \text{chave}[v])$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

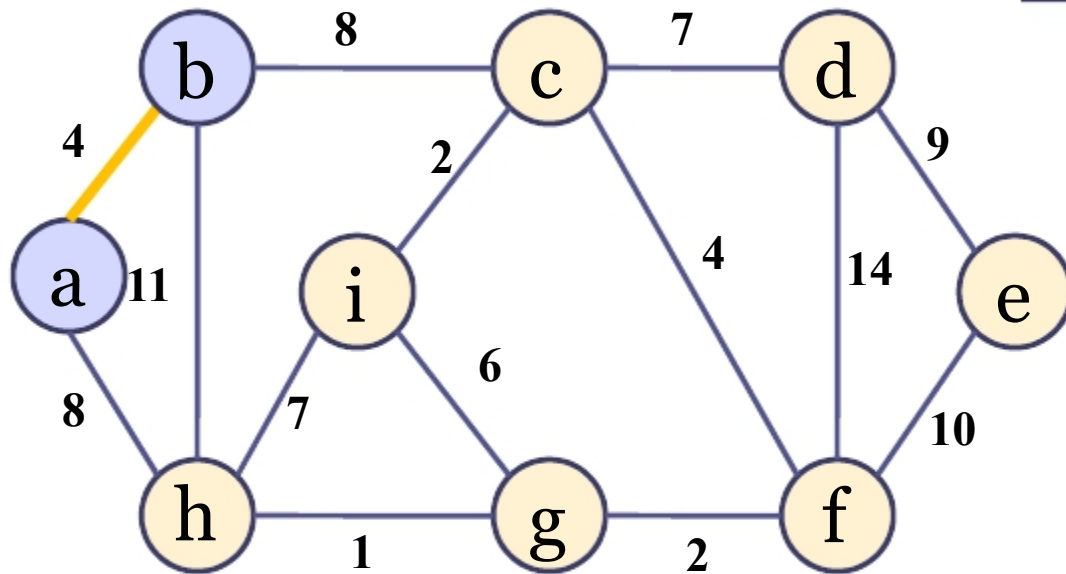
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	∞	∞	∞	∞	∞	8	∞
π	NULL	a	NULL	NULL	NULL	NULL	NULL	a	NULL
Q	---	---	X	X	X	X	X	X	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

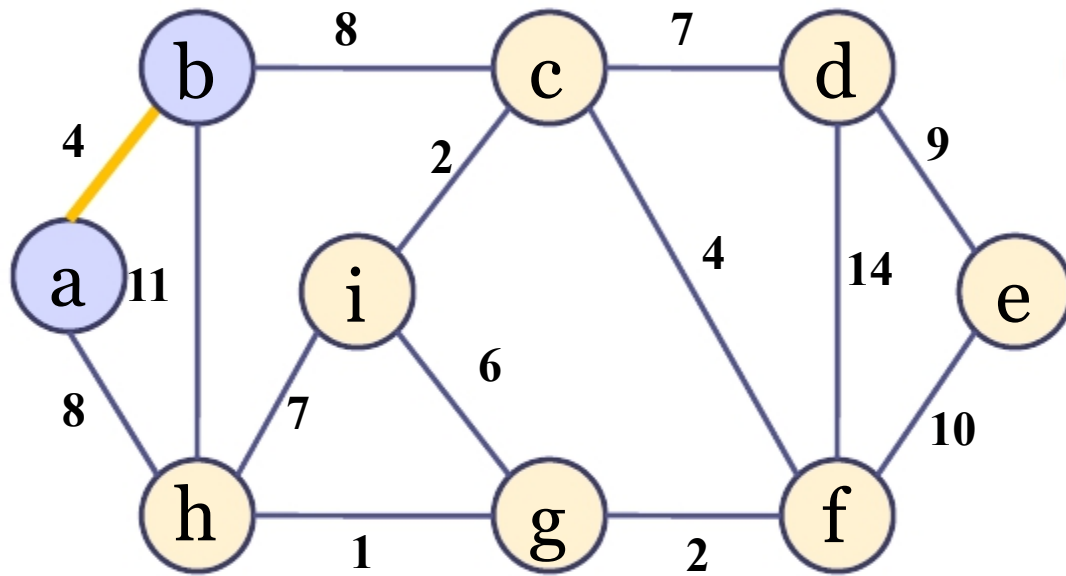
$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

→ para cada $v \in \text{Adj}[u]$ faça
 se $(v \in Q) \text{ e } (w(u, v) < \text{chave}[v])$
 $\text{chave}[v] \leftarrow w(u, v)$
 $\pi[v] \leftarrow u$
 fimse
 fim para
 fim enquanto
 retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	∞	∞	∞	∞	∞	8	∞
π	NULL	a	NULL	NULL	NULL	NULL	NULL	a	NULL
Q	---	---	X	X	X	X	X	X	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

→ se $(v \in Q) \wedge (w(u, v) < \text{chave}[v])$

→ $\text{chave}[v] \leftarrow w(u, v)$

→ $\pi[v] \leftarrow u$

fimse

fim para

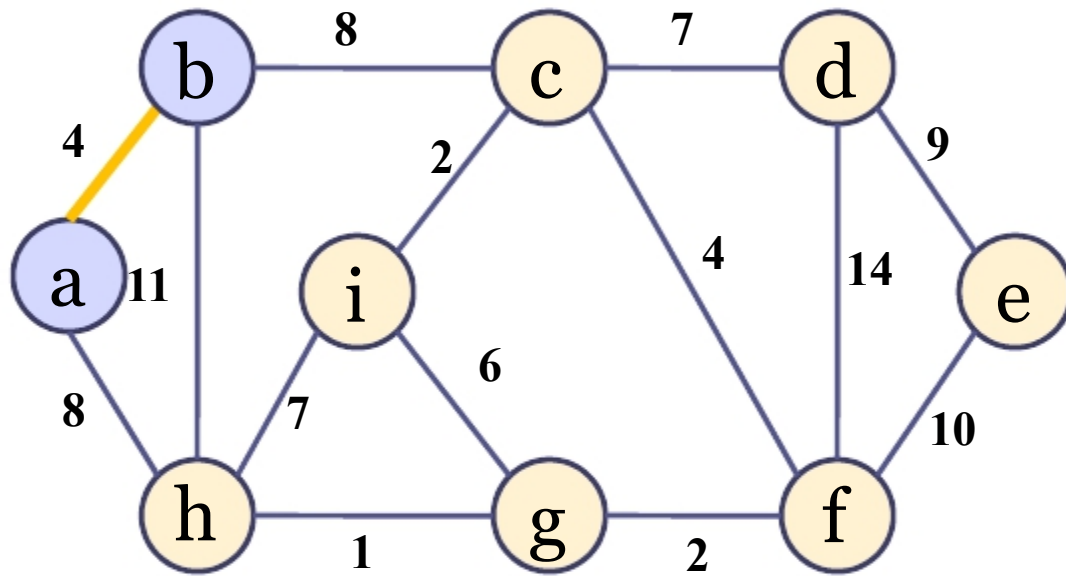
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	8	∞	∞	∞	∞	8	∞
π	NULL	a	b	NULL	NULL	NULL	NULL	a	NULL
Q	---	---	X	X	X	X	X	X	X

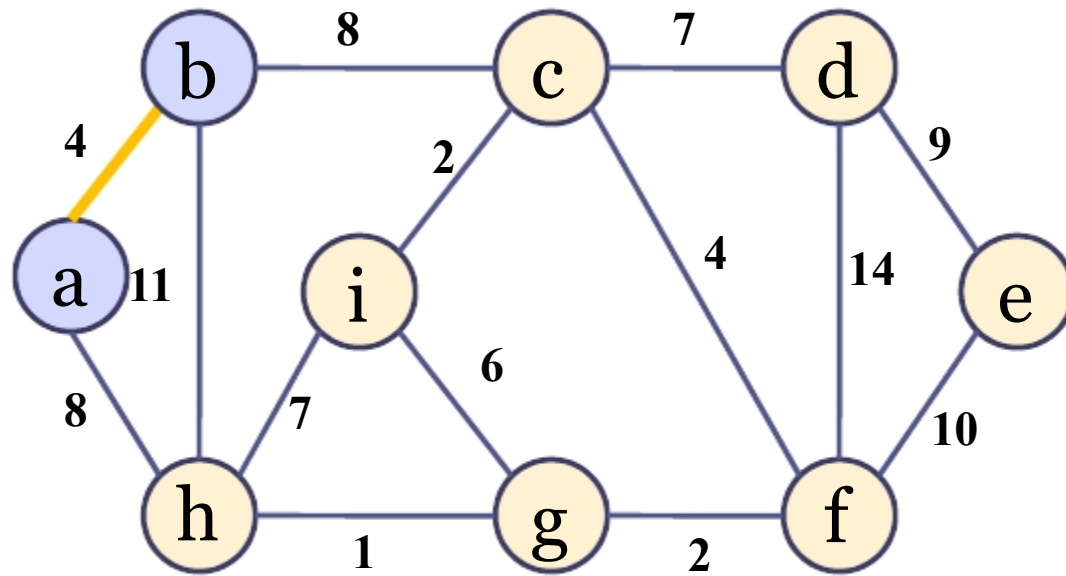
Algoritmo de Prim



enquanto $Q \neq \{\}$
 $u \leftarrow \text{extrairMinimo}(Q)$
 $X \leftarrow X \cup \{(u, \pi[u])\}$ //seu $\neq r$
 para cada $v \in \text{Adj}[u]$ faça
 se $(v \in Q) \wedge (w(u, v) < \text{chave}[v])$
 $\text{chave}[v] \leftarrow w(u, v)$
 $\pi[v] \leftarrow u$
 fimse
 fim para
 fim enquanto
 retorne X
 fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	8	∞	∞	∞	∞	8	∞
π	NULL	a	b	NULL	NULL	NULL	NULL	a	NULL
Q	---	---	X	X	X	X	X	X	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$\rightarrow u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \text{ e } (w(u, v) < \text{chave}[v])$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

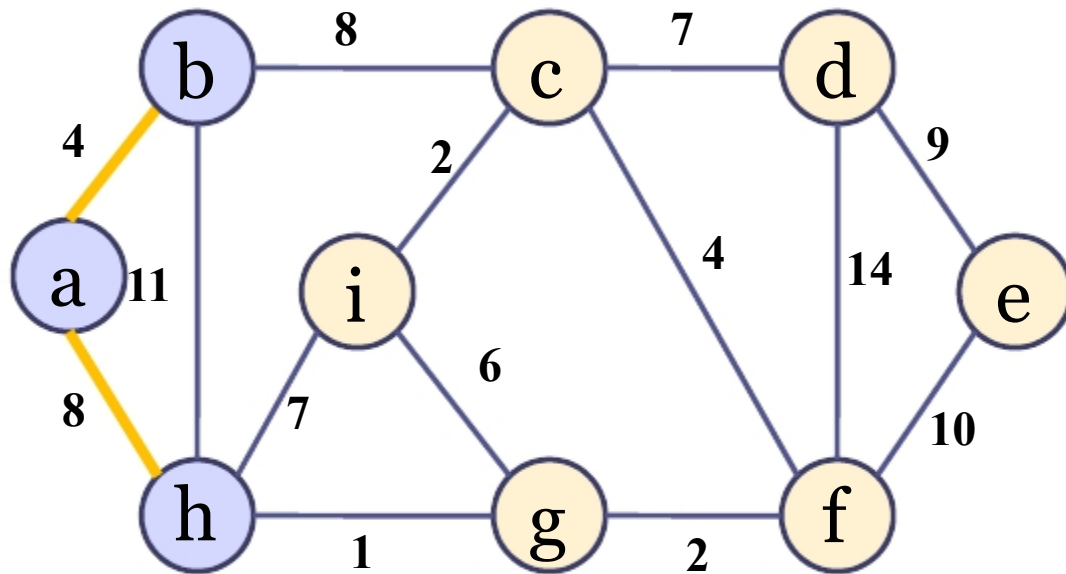
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	8	∞	∞	∞	∞	8	∞
π	NULL	a	b	NULL	NULL	NULL	NULL	a	NULL
Q	---	---	X	X	X	X	X	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

→ $X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \text{ e } (w(u, v) < \text{chave}[v])$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

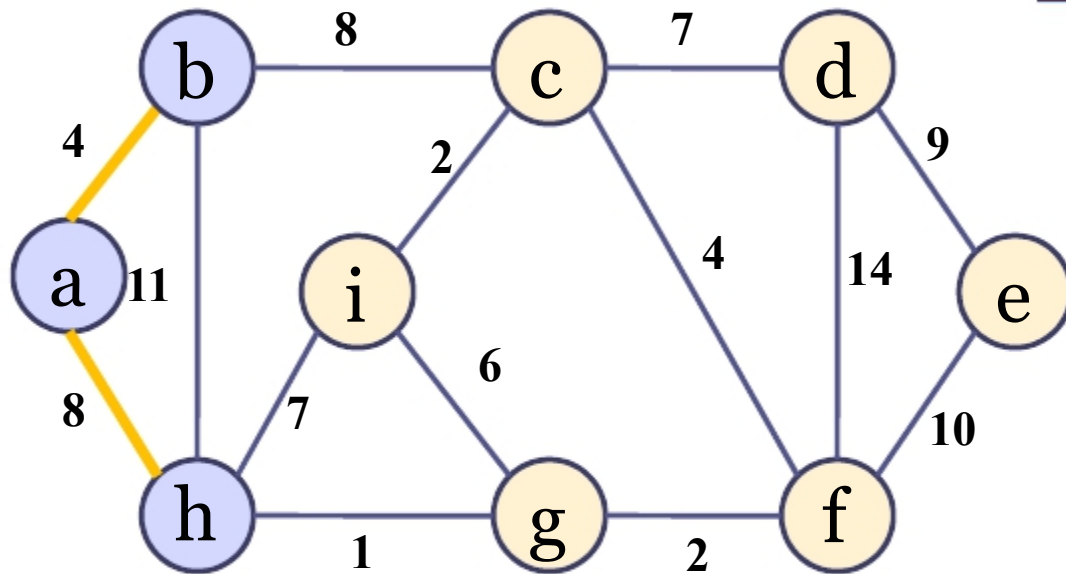
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	8	∞	∞	∞	∞	8	∞
π	NULL	a	b	NULL	NULL	NULL	NULL	a	NULL
Q	---	---	X	X	X	X	X	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

→ para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \text{ e } w(u, v) < \text{chave}[v]$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

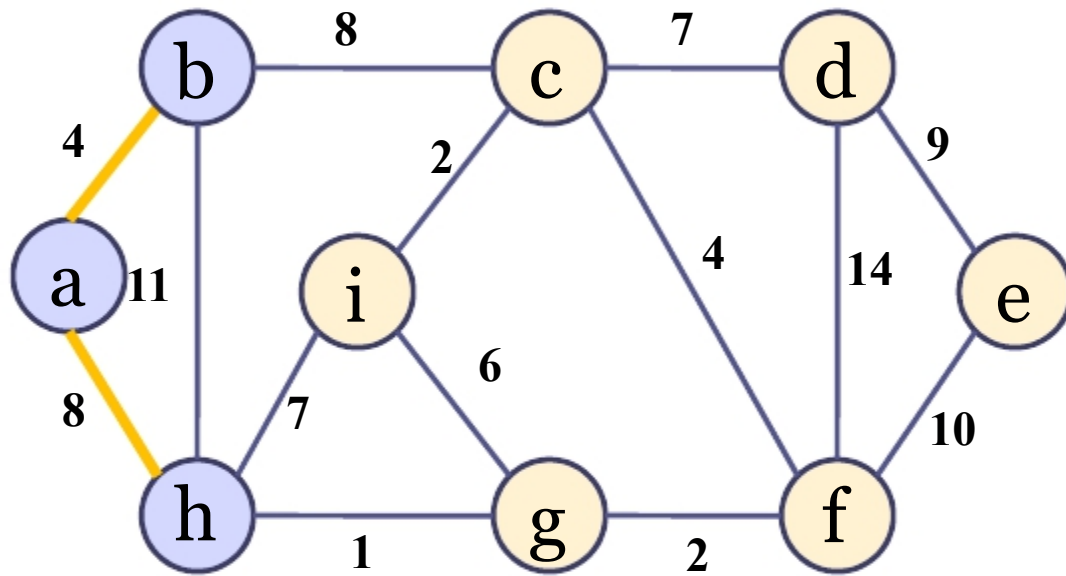
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	8	∞	∞	∞	∞	8	∞
π	NULL	a	b	NULL	NULL	NULL	NULL	a	NULL
Q	---	---	X	X	X	X	X	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\}$ //seu $\neq r$

para cada $v \in \text{Adj}[u]$ faça

→ se $(v \in Q) \text{ e } w(u, v) < \text{chave}[v]$

→ $\text{chave}[v] \leftarrow w(u, v)$

→ $\pi[v] \leftarrow u$

fimse

fim para

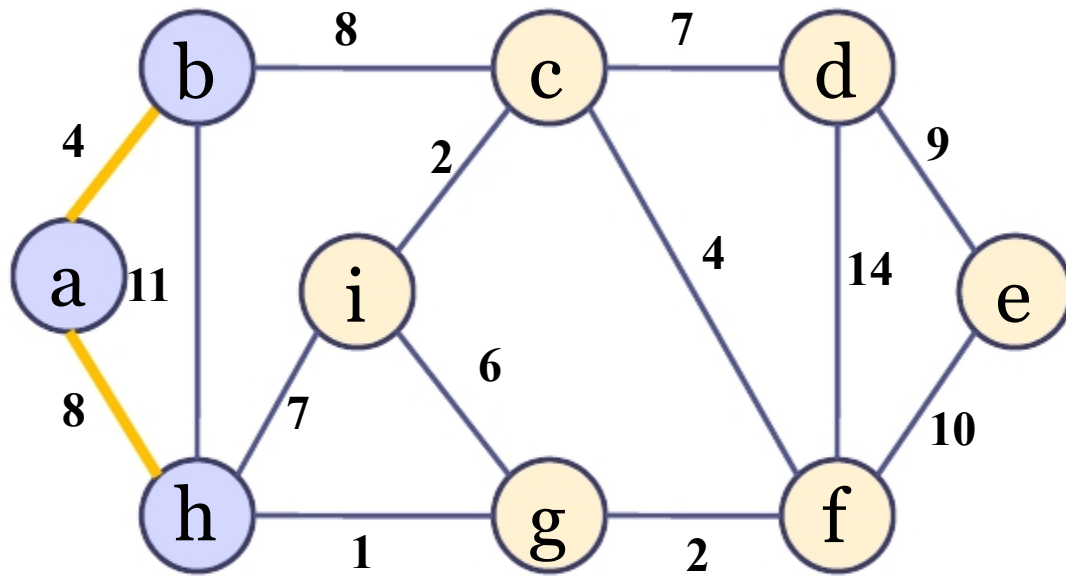
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	8	∞	∞	∞	1	8	7
π	NULL	a	b	NULL	NULL	NULL	h	a	h
Q	---	---	X	X	X	X	X	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \wedge (w(u, v) < \text{chave}[v])$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

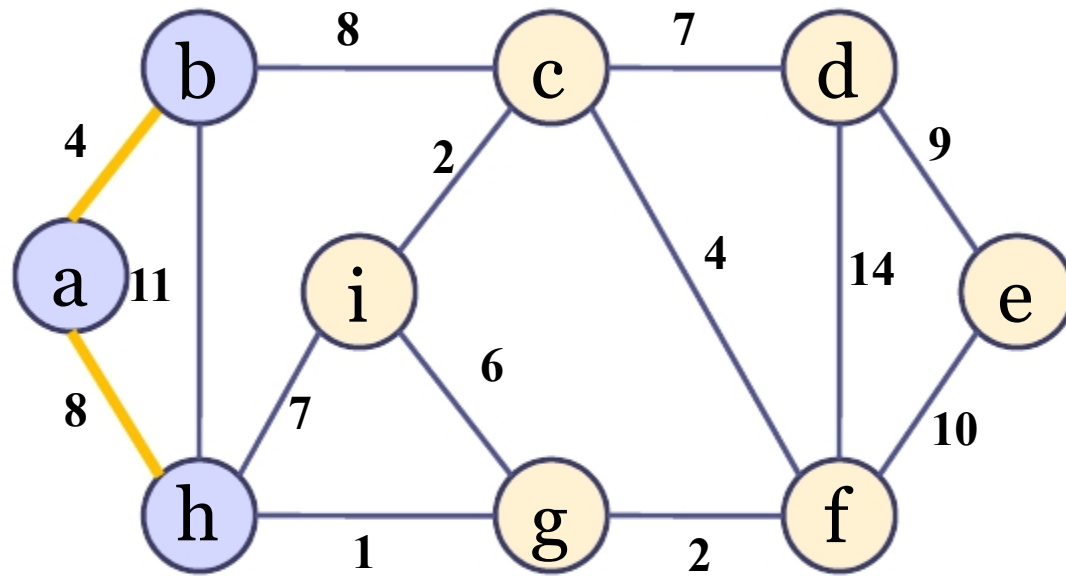
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	8	∞	∞	∞	1	8	7
π	NULL	a	b	NULL	NULL	NULL	h	a	h
Q	---	---	X	X	X	X	X	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$\rightarrow u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \text{ e } w(u, v) < \text{chave}[v]$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

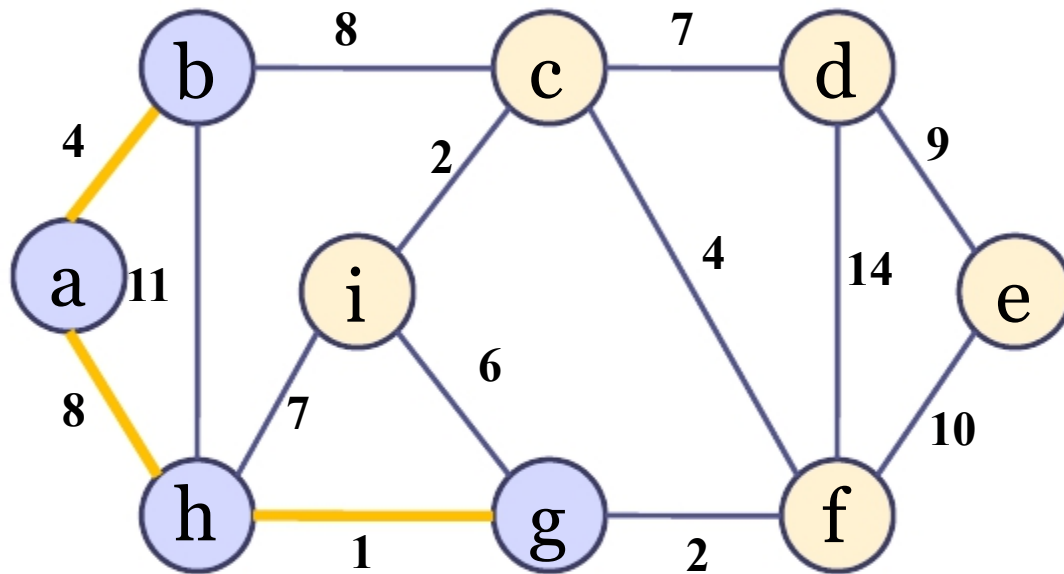
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	8	∞	∞	∞	1	8	7
π	NULL	a	b	NULL	NULL	NULL	h	a	h
Q	---	---	X	X	X	X	---	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

→ $X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \wedge w(u, v) < \text{chave}[v]$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

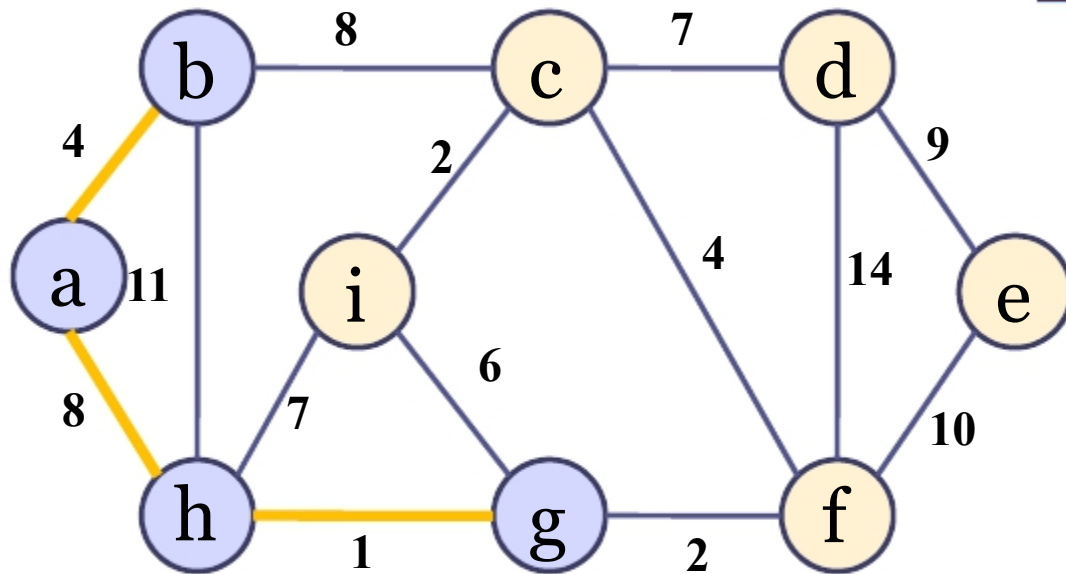
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	8	∞	∞	∞	1	8	7
π	NULL	a	b	NULL	NULL	NULL	h	a	h
Q	---	---	X	X	X	X	---	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

→ para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \text{ e } w(u, v) < \text{chave}[v]$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

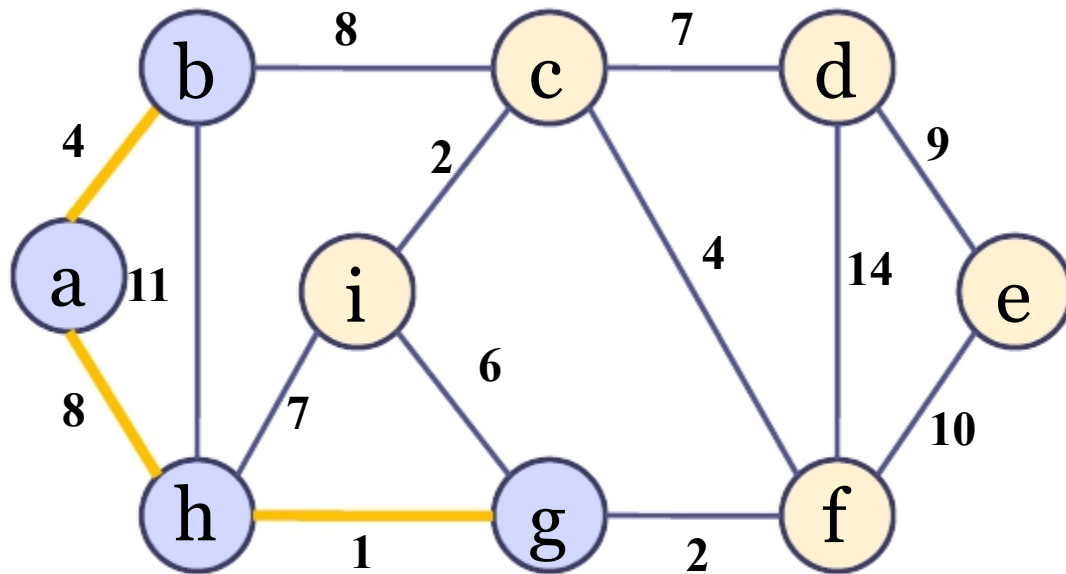
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	8	∞	∞	∞	1	8	7
π	NULL	a	b	NULL	NULL	NULL	h	a	h
Q	---	---	X	X	X	X	---	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

→ se $(v \in Q) \text{ e } w(u, v) < \text{chave}[v]$

→ $\text{chave}[v] \leftarrow w(u, v)$

→ $\pi[v] \leftarrow u$

fimse

fim para

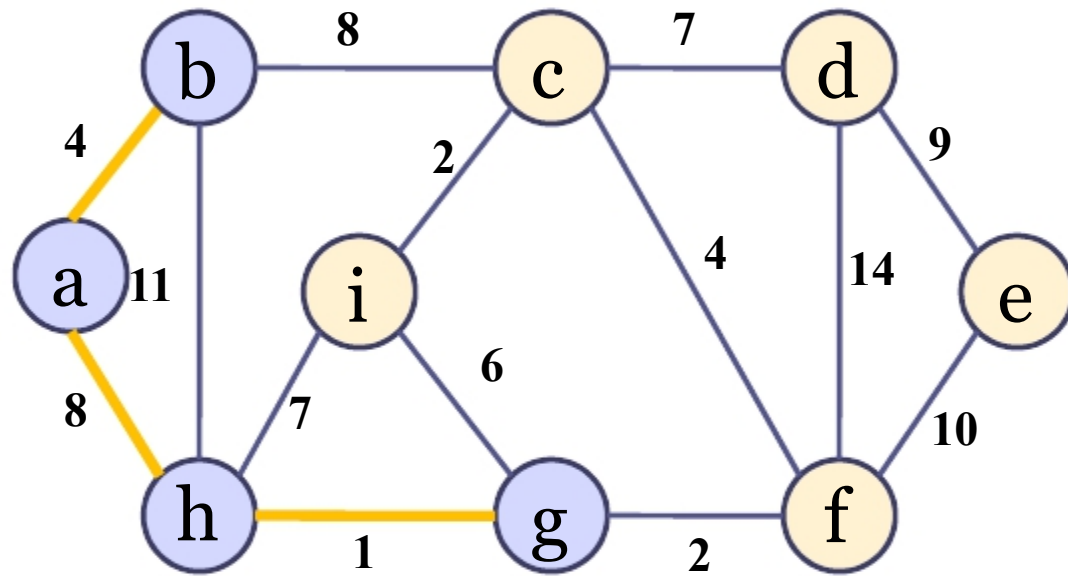
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	8	∞	∞	2	1	8	6
π	NULL	a	b	NULL	NULL	g	h	a	g
Q	---	---	X	X	X	X	---	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \text{ e } w(u, v) < \text{chave}[v]$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

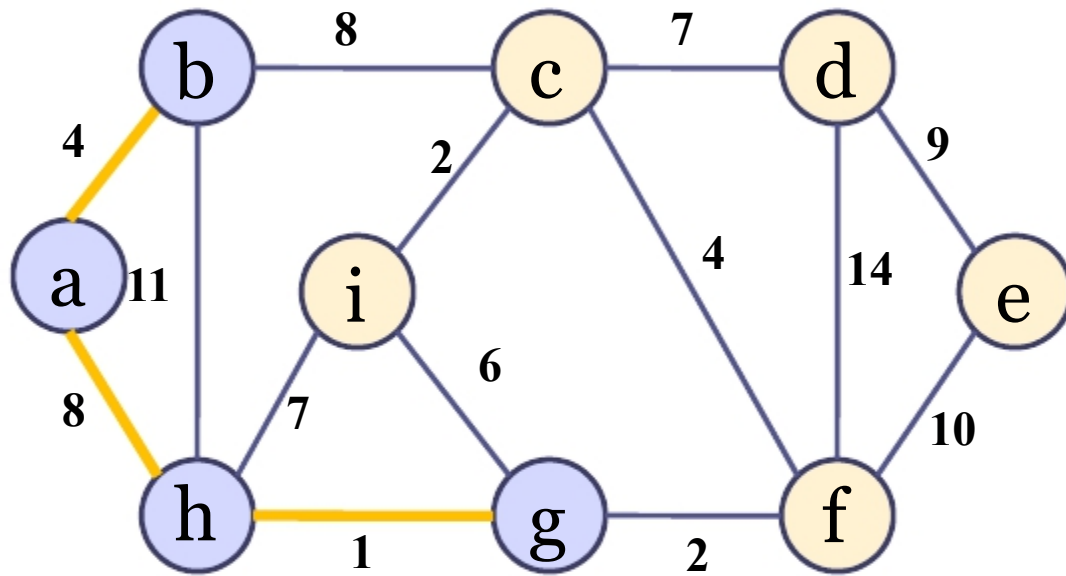
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	8	∞	∞	2	1	8	6
π	NULL	a	b	NULL	NULL	g	h	a	g
Q	---	---	X	X	X	X	---	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$\rightarrow u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \text{ e } w(u, v) < \text{chave}[v]$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

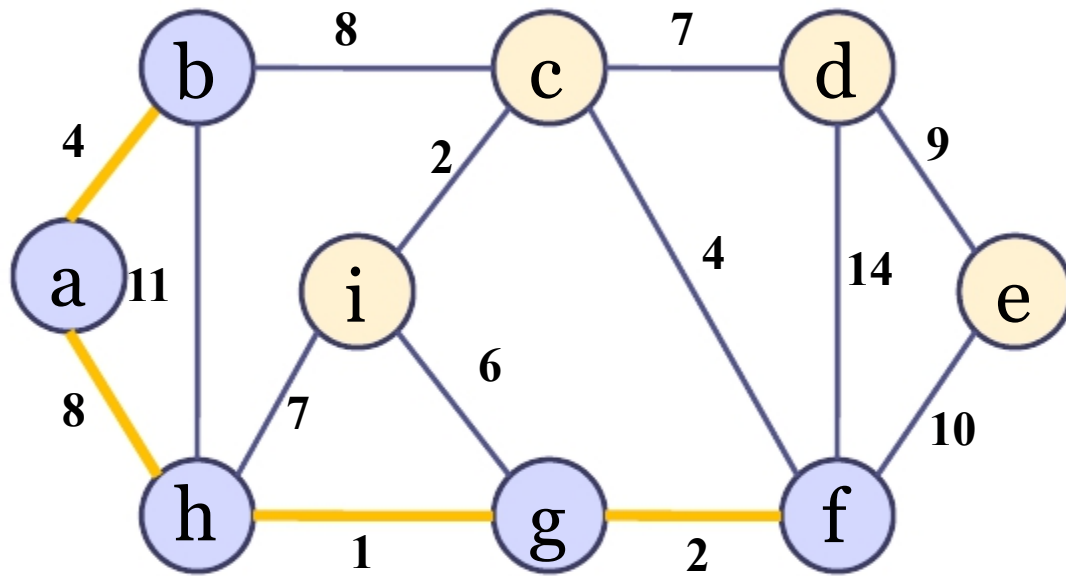
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	8	∞	∞	2	1	8	6
π	NULL	a	b	NULL	NULL	g	h	a	g
Q	---	---	X	X	X	---	---	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

→ $X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \wedge w(u, v) < \text{chave}[v]$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

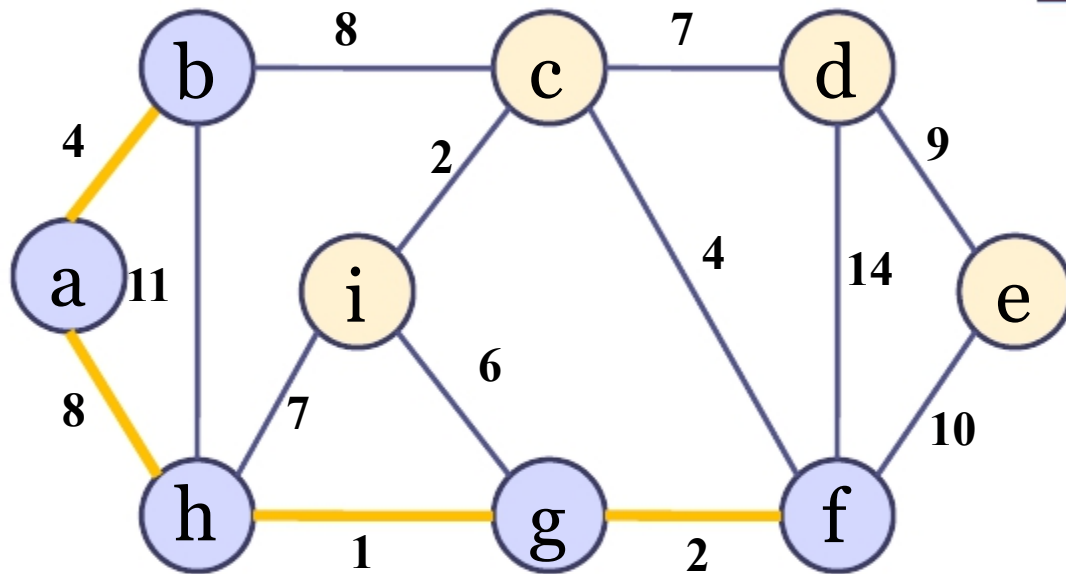
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	8	∞	∞	2	1	8	6
π	NULL	a	b	NULL	NULL	g	h	a	g
Q	---	---	X	X	X	---	---	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

→ para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \text{ e } (w(u, v) < \text{chave}[v])$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

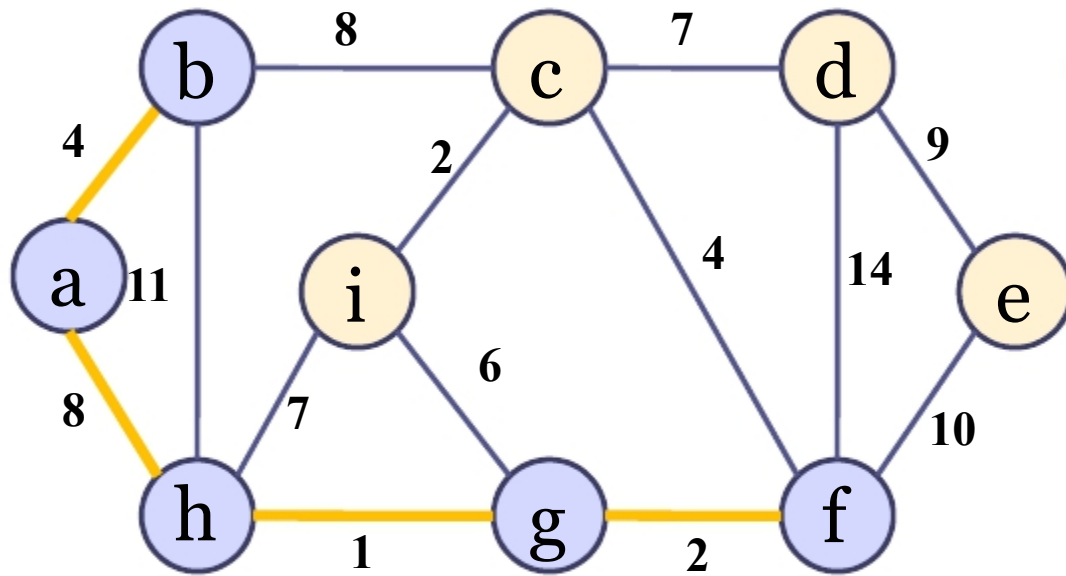
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	8	∞	∞	2	1	8	6
π	NULL	a	b	NULL	NULL	g	h	a	g
Q	---	---	X	X	X	---	---	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

→ se $(v \in Q) \wedge (w(u, v) < \text{chave}[v])$

→ $\text{chave}[v] \leftarrow w(u, v)$

→ $\pi[v] \leftarrow u$

fimse

fim para

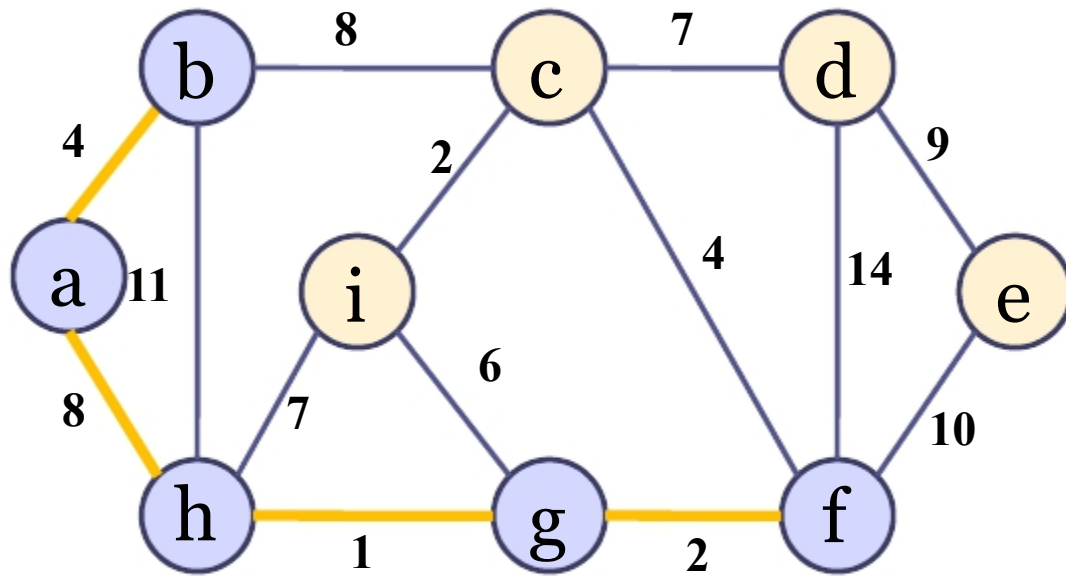
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	4	14	10	2	1	8	6
π	NULL	a	f	f	f	g	h	a	g
Q	---	---	X	X	X	---	---	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \vee (w(u, v) < \text{chave}[v])$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

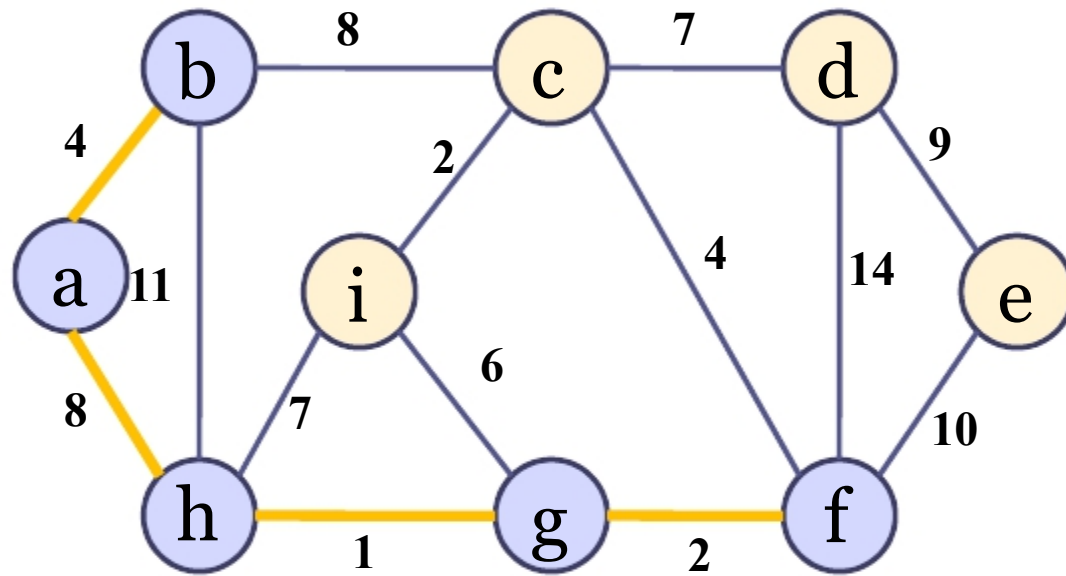
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	4	14	10	2	1	8	6
π	NULL	a	f	f	f	g	h	a	g
Q	---	---	X	X	X	---	---	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$



$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \text{ e } (w(u, v) < \text{chave}[v])$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

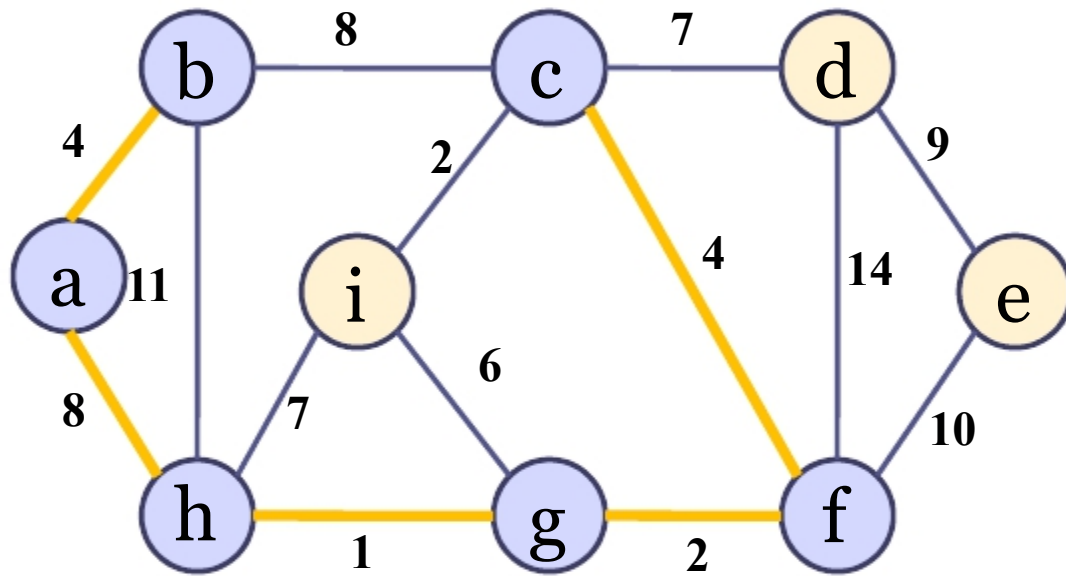
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	4	14	10	2	1	8	6
π	NULL	a	f	f	f	g	h	a	g
Q	---	---	---	X	X	---	---	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

→ $X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \wedge (w(u, v) < \text{chave}[v])$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

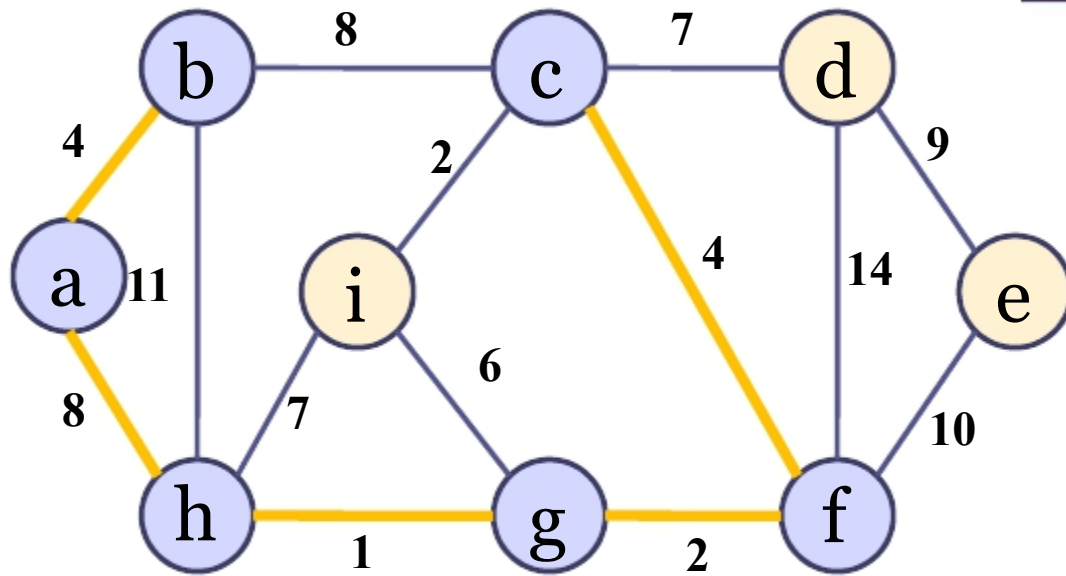
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	4	14	10	2	1	8	6
π	NULL	a	f	f	f	g	h	a	g
Q	---	---	---	X	X	---	---	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

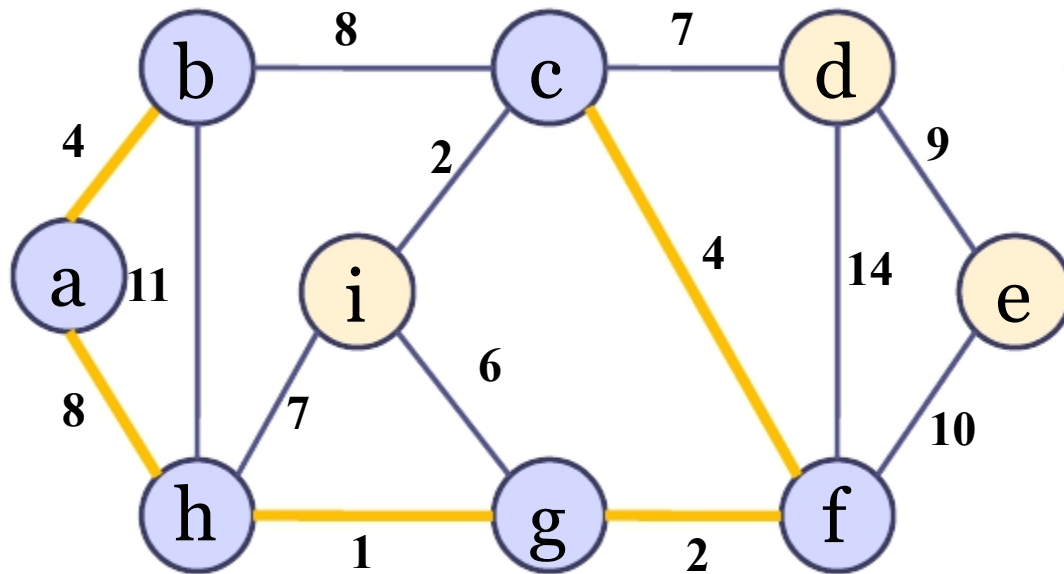
$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

→ para cada $v \in \text{Adj}[u]$ faça
 se $(v \in Q) \text{ e } w(u, v) < \text{chave}[v]$
 $\text{chave}[v] \leftarrow w(u, v)$
 $\pi[v] \leftarrow u$
 fimse
 fim para
 fim enquanto
 retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	4	14	10	2	1	8	6
π	NULL	a	f	f	f	g	h	a	g
Q	---	---	---	X	X	---	---	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

→ se $(v \in Q) \vee (w(u, v) < \text{chave}[v])$

→ $\text{chave}[v] \leftarrow w(u, v)$

→ $\pi[v] \leftarrow u$

fimse

fim para

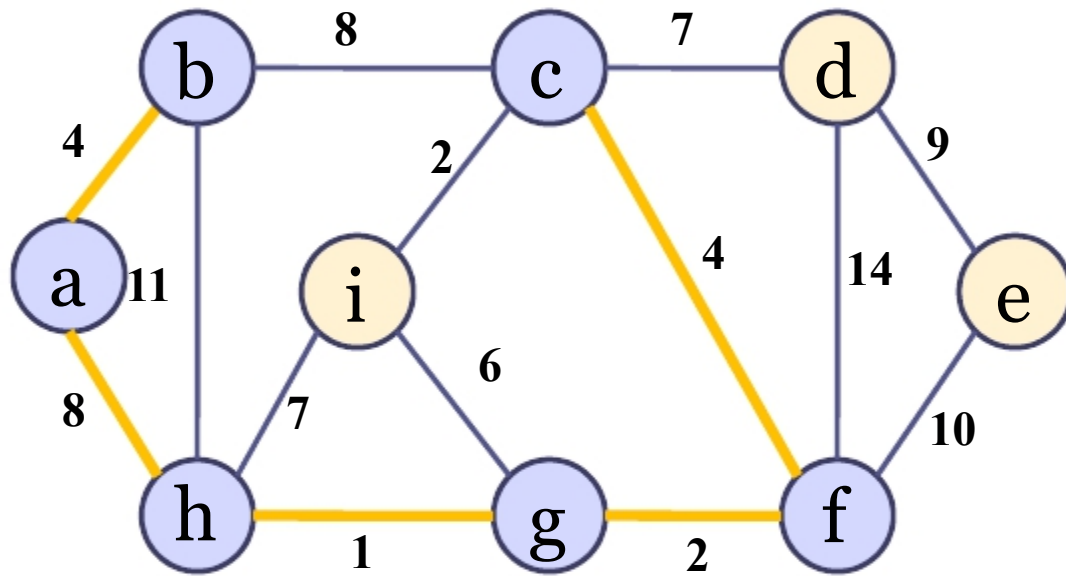
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	4	7	10	2	1	8	2
π	NULL	a	f	c	f	g	h	a	c
Q	---	---	---	X	X	---	---	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \wedge (w(u, v) < \text{chave}[v])$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

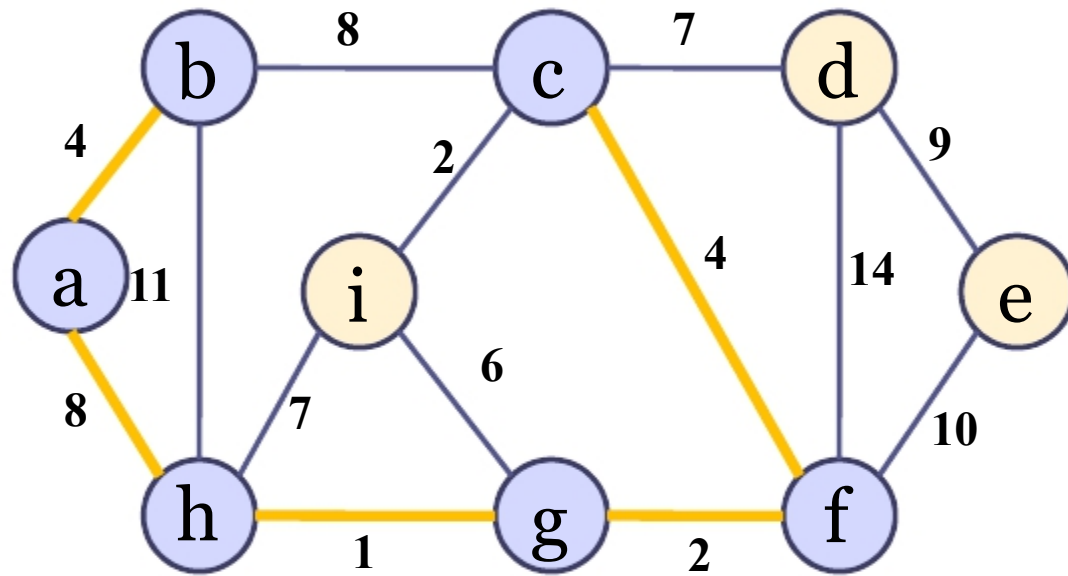
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	4	7	10	2	1	8	2
π	NULL	a	f	c	f	g	h	a	c
Q	---	---	---	X	X	---	---	---	X

Algoritmo de Prim



enquanto $Q \neq \{\}$

$\rightarrow u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \wedge (w(u, v) < \text{chave}[v])$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

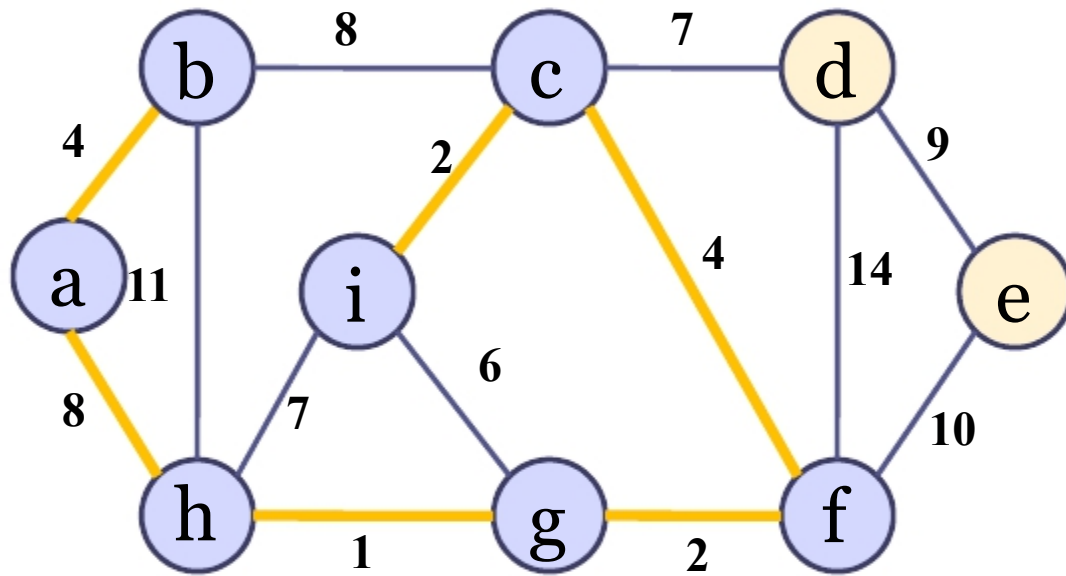
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	4	7	10	2	1	8	2
π	NULL	a	f	c	f	g	h	a	c
Q	---	---	---	X	X	---	---	---	---

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

→ $X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \wedge (w(u, v) < \text{chave}[v])$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

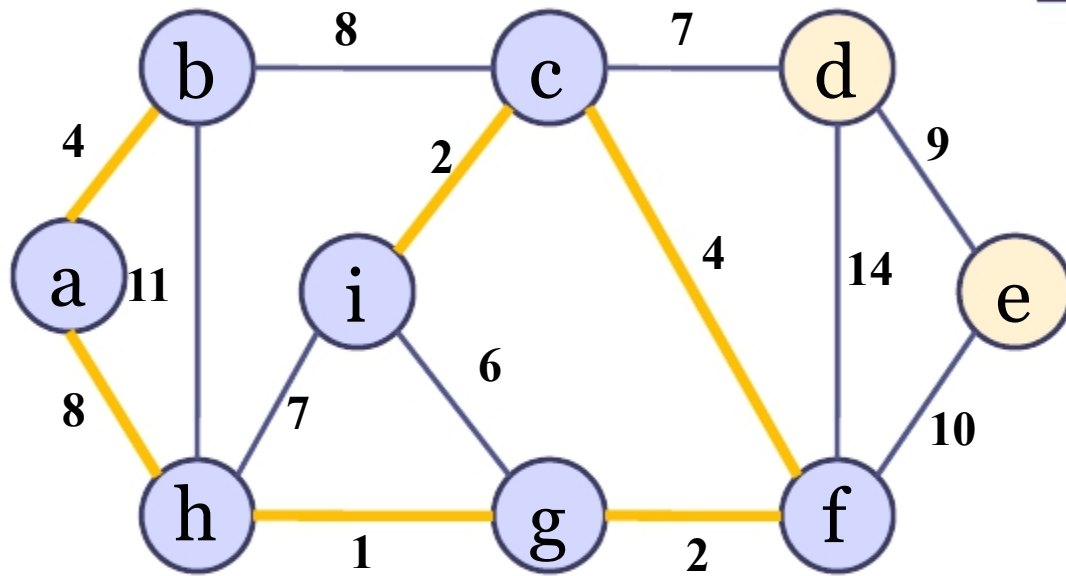
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	4	7	10	2	1	8	2
π	NULL	a	f	c	f	g	h	a	c
Q	---	---	---	X	X	---	---	---	---

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

➔ para cada $v \in \text{Adj}[u]$ faça

➔ se $(v \in Q) \wedge (w(u, v) < \text{chave}[v])$

➔ $\text{chave}[v] \leftarrow w(u, v)$

➔ $\pi[v] \leftarrow u$

fimse

fim para

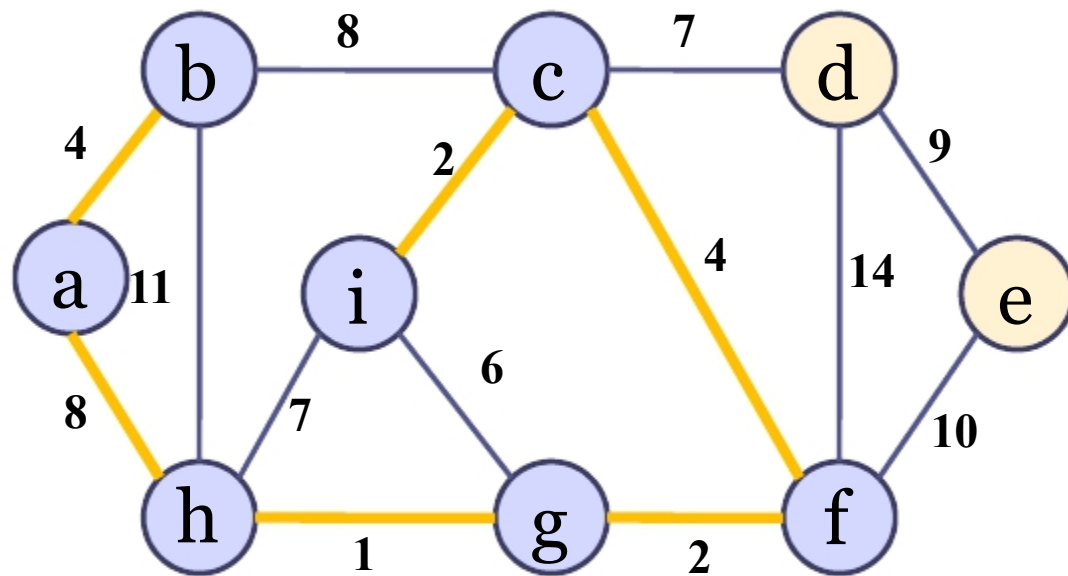
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	4	7	10	2	1	8	2
π	NULL	a	f	c	f	g	h	a	c
Q	---	---	---	X	X	---	---	---	---

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \wedge (w(u, v) < \text{chave}[v])$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

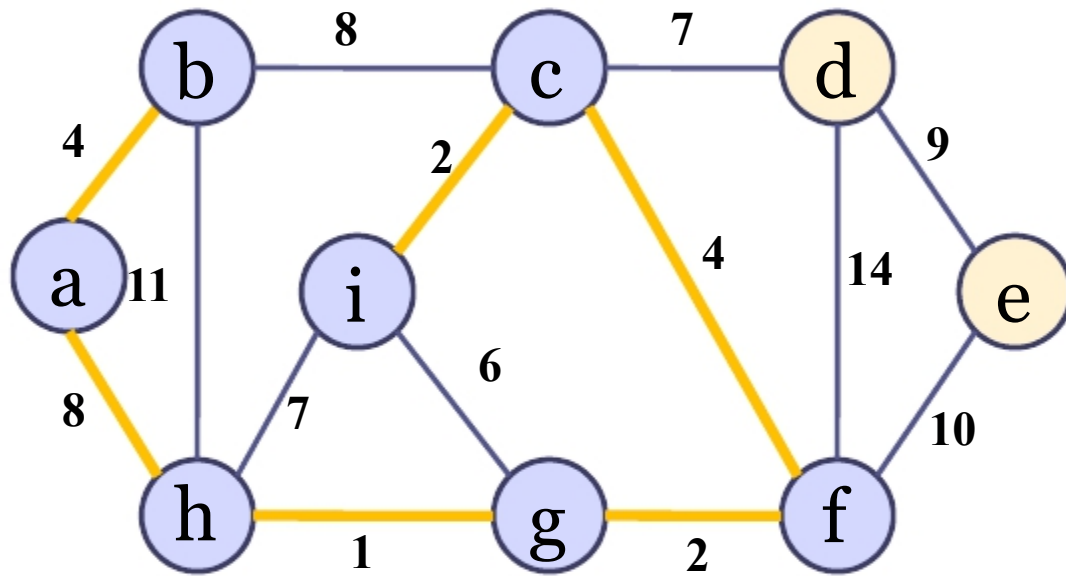
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	4	7	10	2	1	8	2
π	NULL	a	f	c	f	g	h	a	c
Q	---	---	---	X	X	---	---	---	---

Algoritmo de Prim



enquanto $Q \neq \{\}$

$\rightarrow u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \vee (w(u, v) < \text{chave}[v])$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

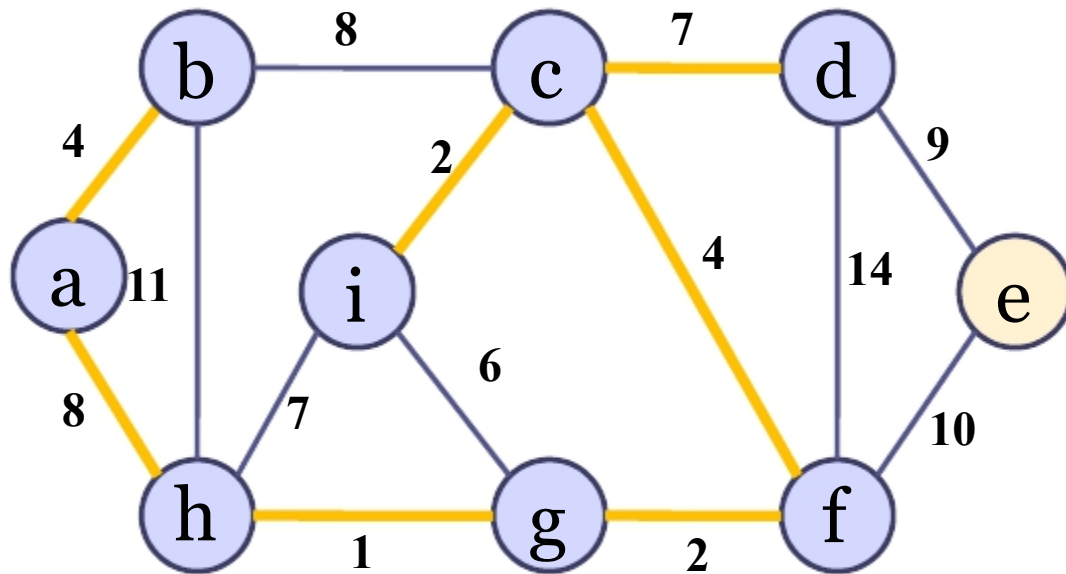
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	4	7	10	2	1	8	2
π	NULL	a	f	c	f	g	h	a	c
Q	---	---	---	---	X	---	---	---	---

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

→ $X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \wedge (w(u, v) < \text{chave}[v])$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

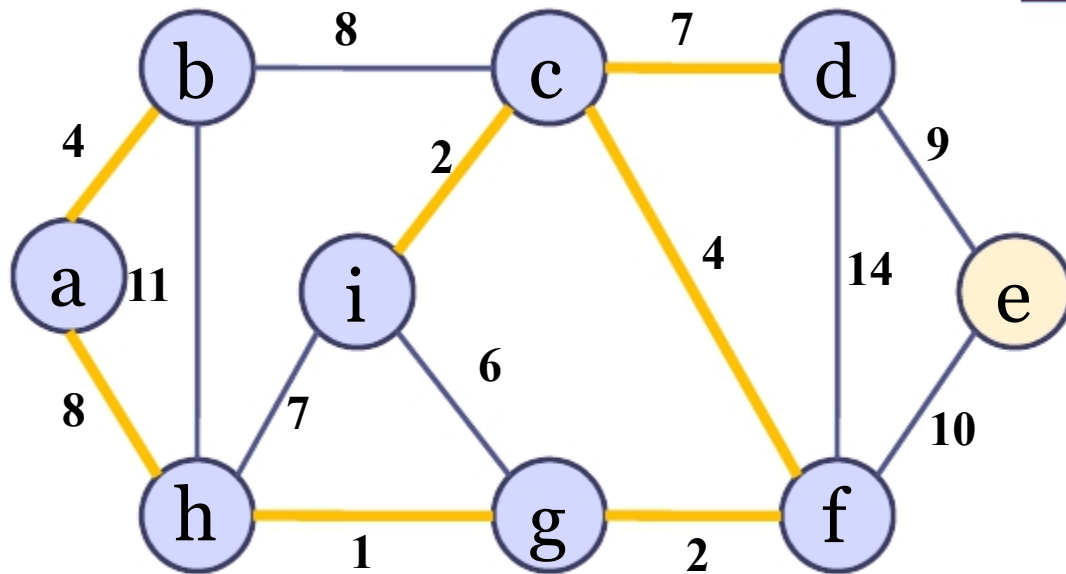
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	4	7	10	2	1	8	2
π	NULL	a	f	c	f	g	h	a	c
Q	---	---	---	---	X	---	---	---	---

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$



para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \text{ e } (w(u, v) < \text{chave}[v])$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

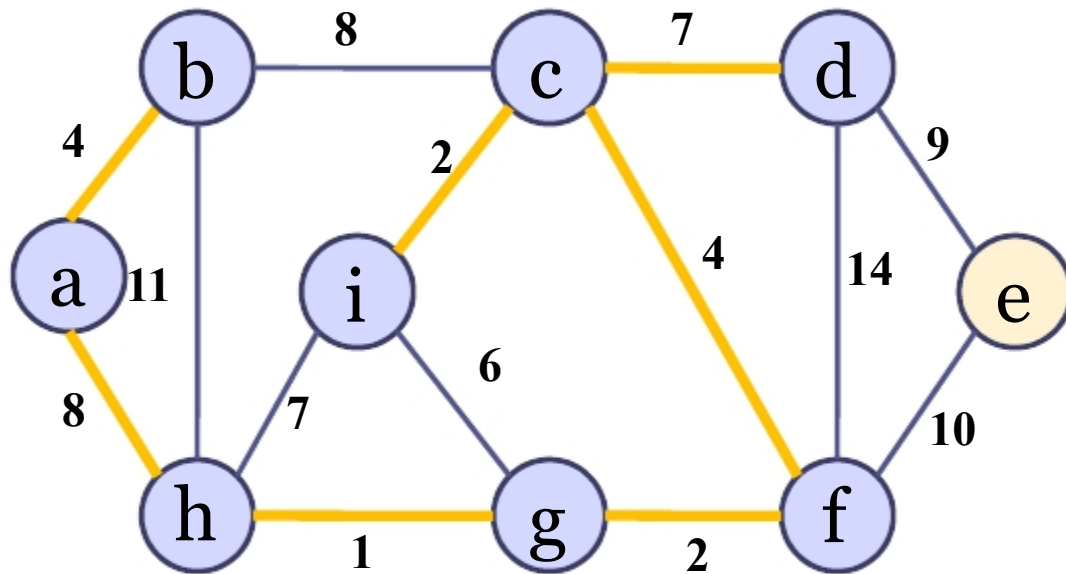
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	4	7	10	2	1	8	2
π	NULL	a	f	c	f	g	h	a	c
Q	---	---	---	---	X	---	---	---	---

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

→ se $(v \in Q) \text{ e } w(u, v) < \text{chave}[v]$

→ $\text{chave}[v] \leftarrow w(u, v)$

→ $\pi[v] \leftarrow u$

fimse

fim para

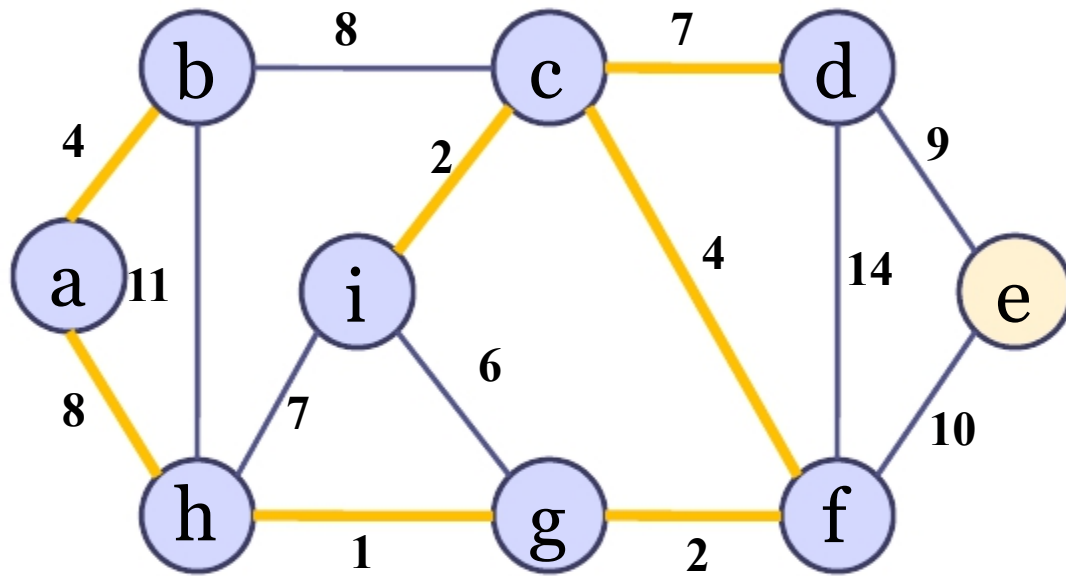
fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	4	7	9	2	1	8	2
π	NULL	a	f	c	d	g	h	a	c
Q	---	---	---	---	X	---	---	---	---

Algoritmo de Prim



enquanto $Q \neq \{\}$

$u \leftarrow \text{extrairMinimo}(Q)$

$X \leftarrow X \cup \{(u, \pi[u])\} // \text{seu} \neq r$

para cada $v \in \text{Adj}[u]$ faça

se $(v \in Q) \vee (w(u, v) < \text{chave}[v])$

$\text{chave}[v] \leftarrow w(u, v)$

$\pi[v] \leftarrow u$

fimse

fim para

fimenquanto

retorne X

fim.

vértice	a	b	c	d	e	f	g	h	i
chave	0	4	4	7	9	2	1	8	2
π	NULL	a	f	c	d	g	h	a	c
Q	---	---	---	---	X	---	---	---	---

enquanto $Q \neq \{\}$

 $u \leftarrow \text{extrairMinimo}(Q)$

$$X \leftarrow X \cup \{(u, \pi[u])\} // seu \neq r$$

para cada $v \in Adj[u]$ faça

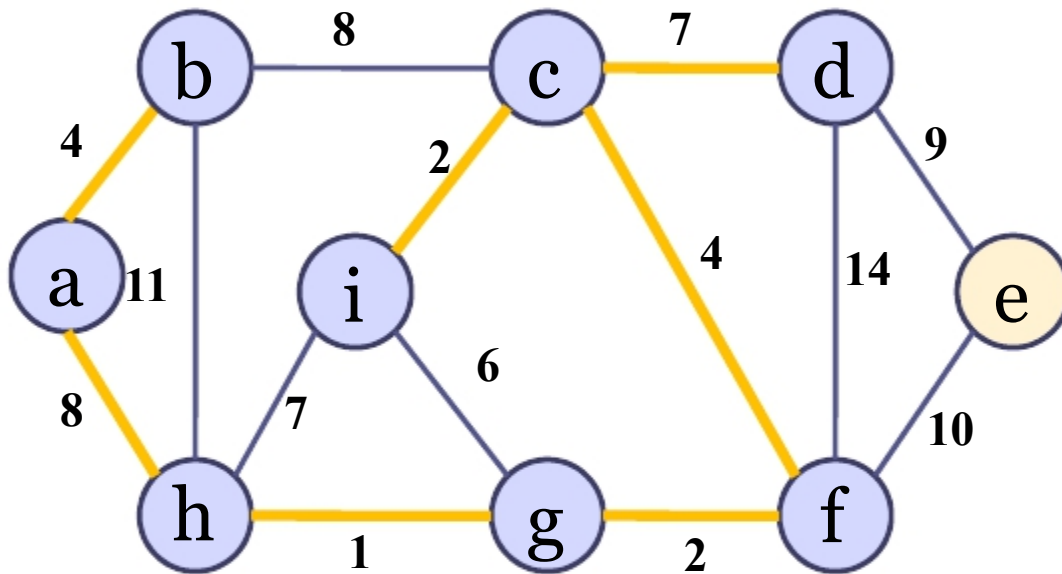
$$se (v \in Q) e(w(u, v) < chav[e][v])$$
$$chave[v] \leftarrow w(u, v)$$
$$\pi[v] \leftarrow u$$

fimse

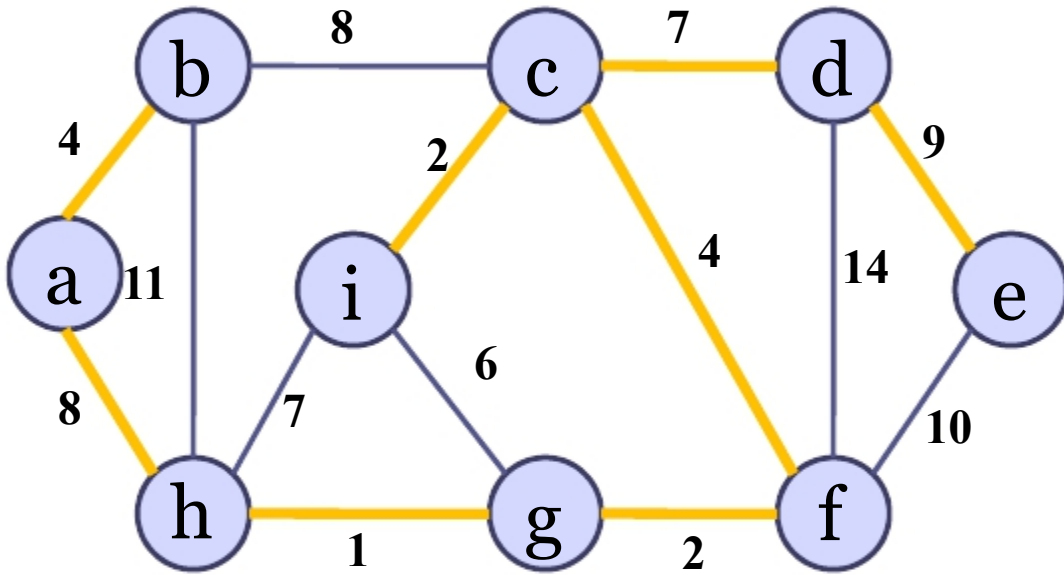
fim para

fimenquanto

retorne X

[illegible]

Algoritmo de Prim



enquanto $Q \neq \{\}$

$$u \leftarrow \text{extrairMinimo}(Q)$$

→ $X \leftarrow X \cup \{(u, \pi[u])\} // seu \neq r$

para cada $v \in Adj[u]$ faça

$$se (v \in Q) e(w(u, v) < chav[e[v]])$$
$$chave[v] \leftarrow w(u, v)$$
$$\pi[v] \leftarrow u$$

fimse

fim para

fimenquanto

retorne X

fim.

[illegible]

$$u \leftarrow \text{extrairMinimo}(Q)$$

 para cada $v \in Adj[u]$ faça

→ $chave[v] \leftarrow w(u, v)$

fimse

fimenquanto

fim.

[illegible]

$$u \leftarrow \text{extrairMinimo}(Q)$$

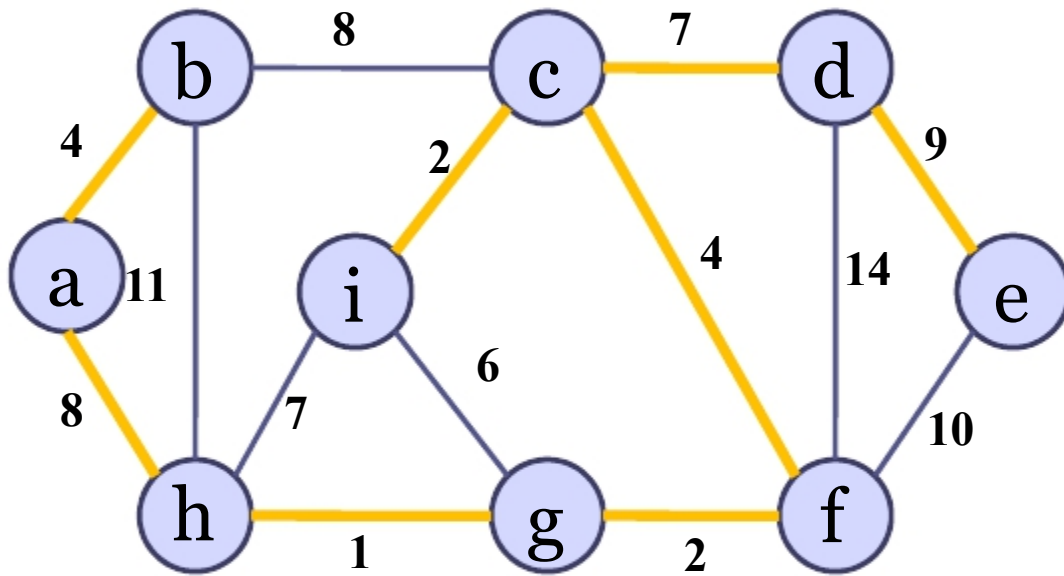
para cada $v \in Adj[u]$ faça

$$chave[v] \leftarrow w(u, v)$$

fimse

fimenquanto

fim.

[illegible]

$$u \leftarrow \text{extrairMinimo}(Q)$$

para cada $v \in Adj[u]$ faça

$$chave[v] \leftarrow w(u, v)$$

fimse

fimenquanto



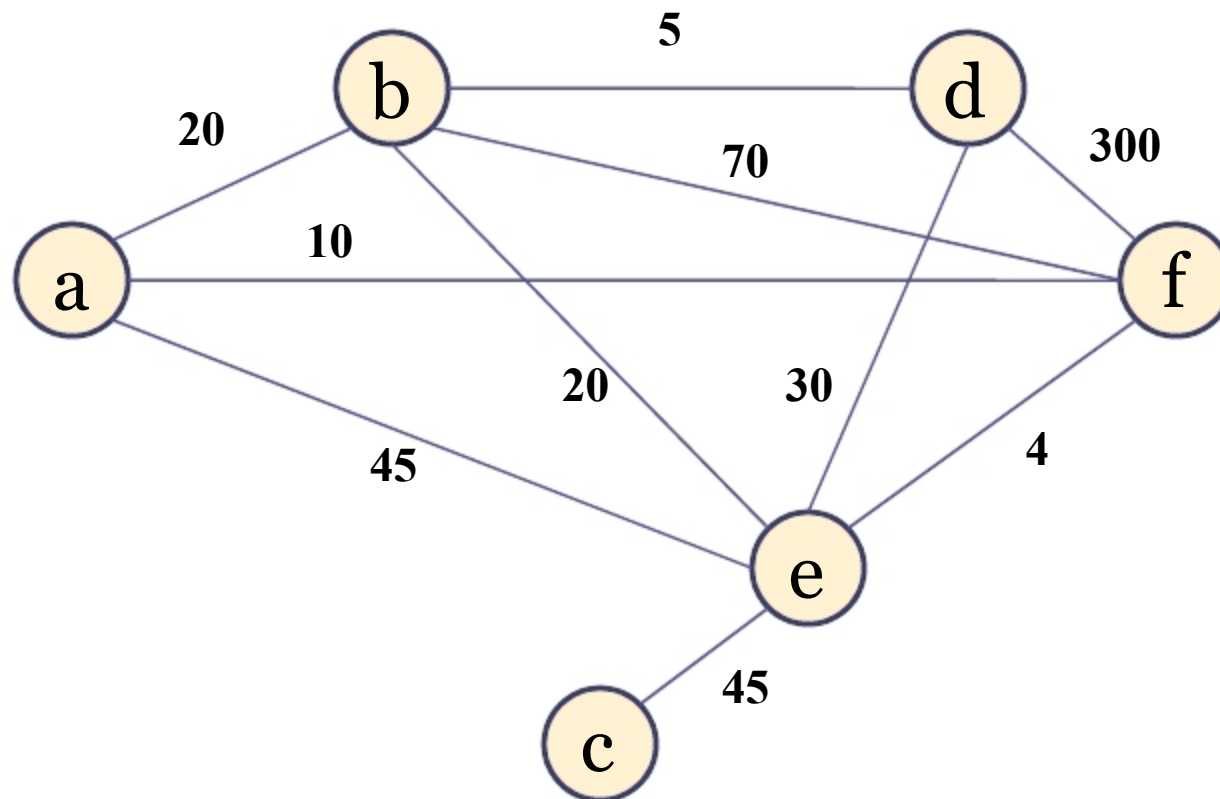
fim.

[illegible]

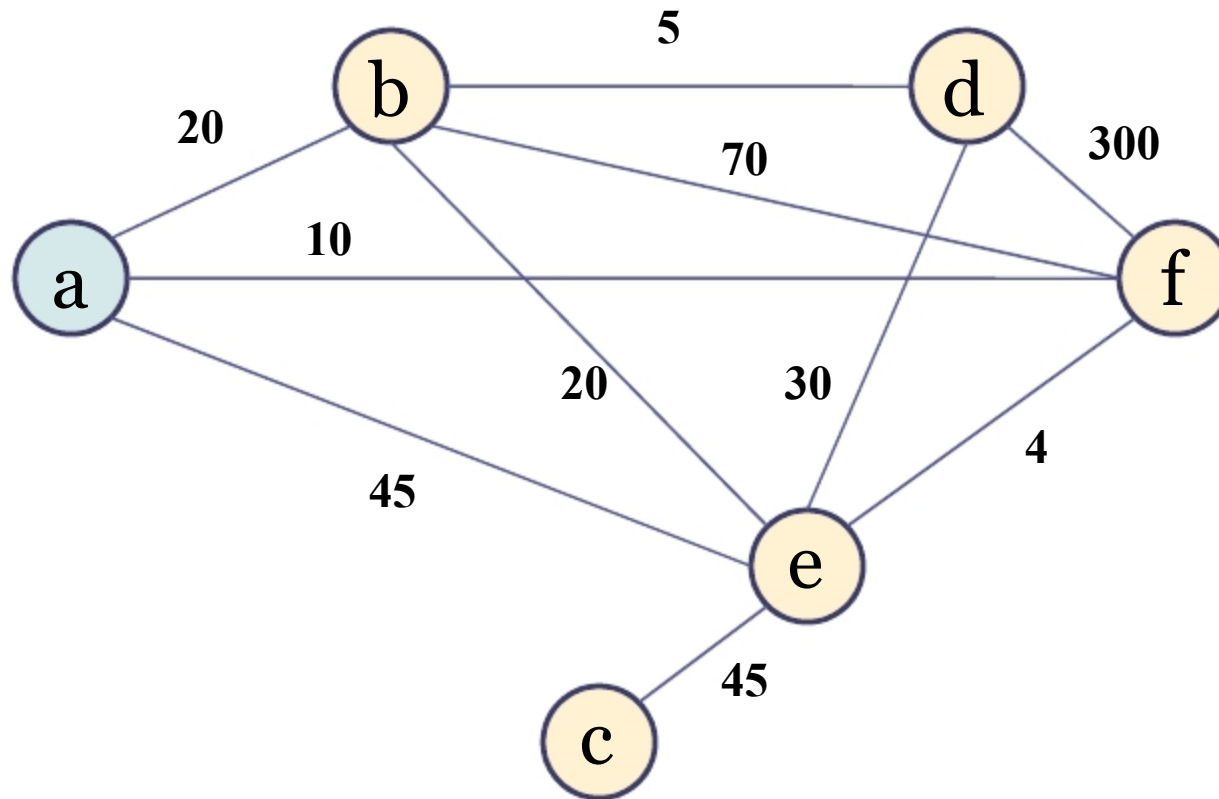
Entendendo PRIM nos moldes do Algoritmo Genérico

Aplique PRIM no seguinte grafo

- AGM_PRIM((V,A), w, a)
 - Começando do vértice a;



Aplique PRIM no seguinte grafo

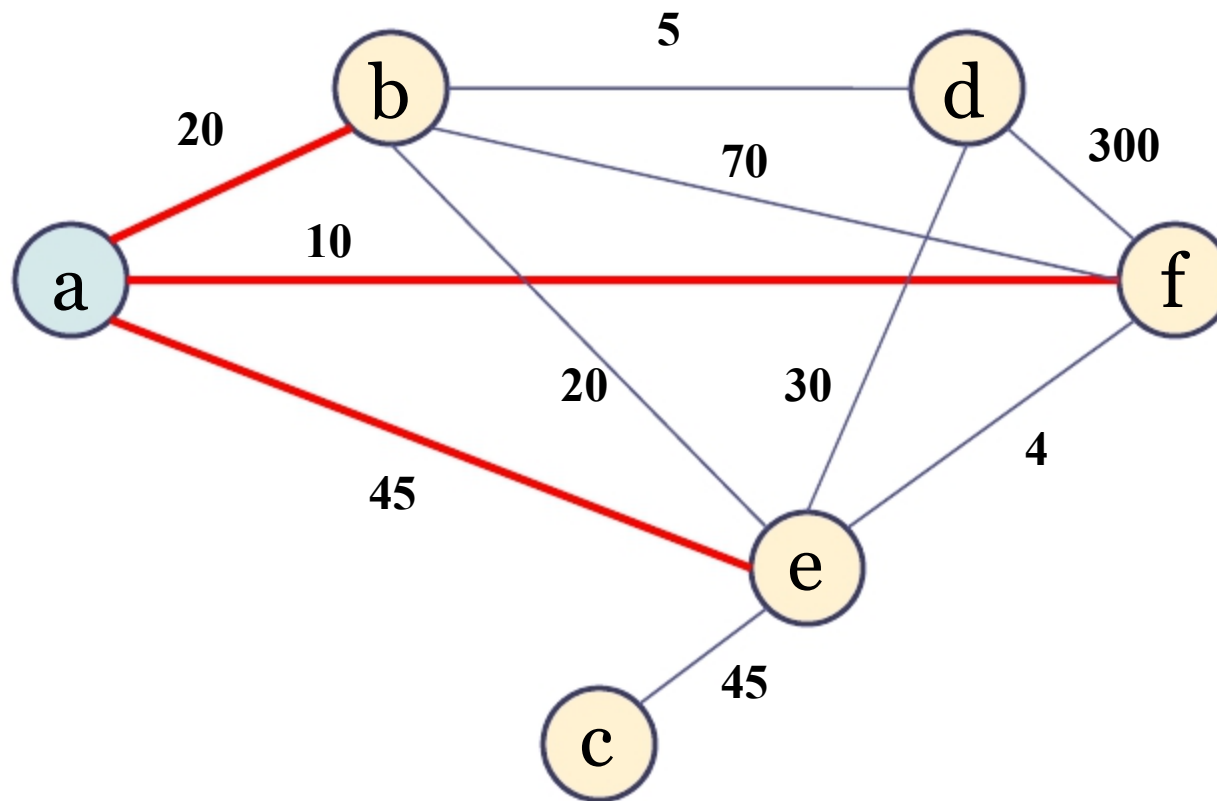


Conjunto S: {a}

Conjunto V-S: {b,c,d,e,f}

X={ }

Aplique PRIM no seguinte grafo

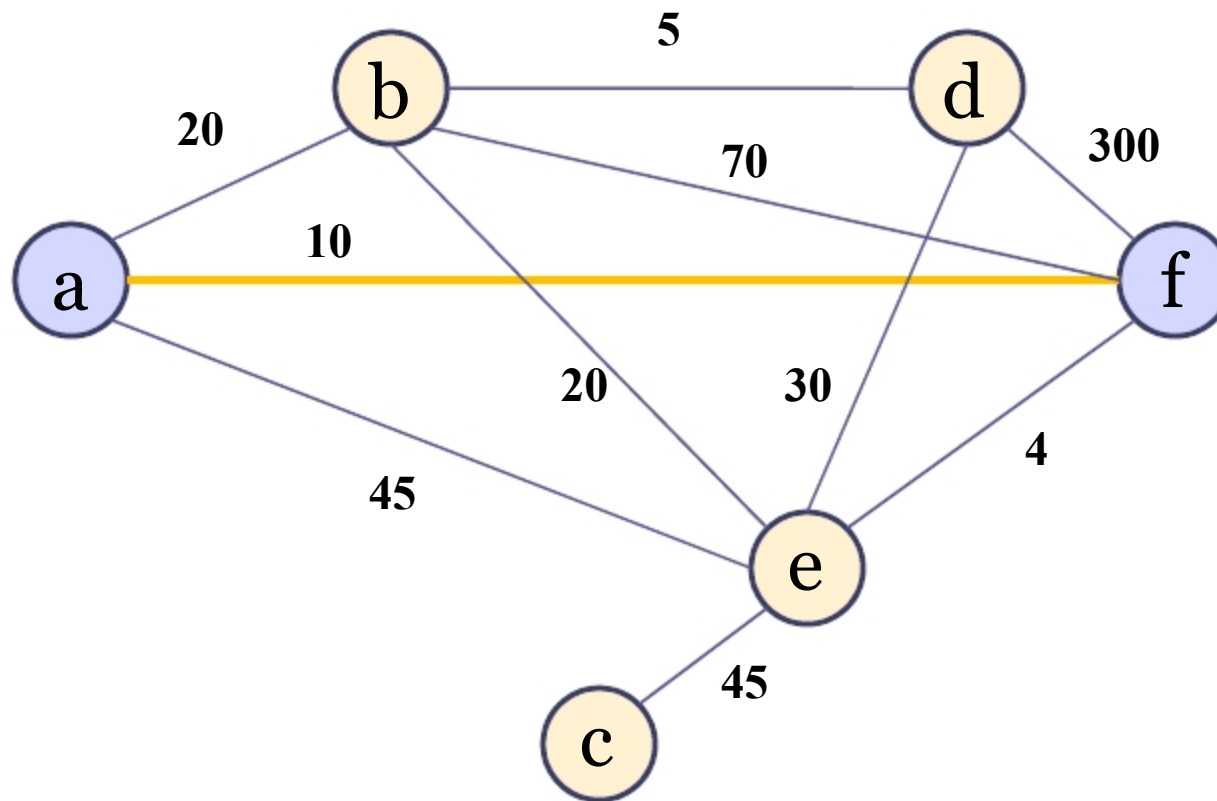


Conjunto S: {a}

Conjunto V-S: {b,c,d,e,f}

X = { }

Aplique PRIM no seguinte grafo

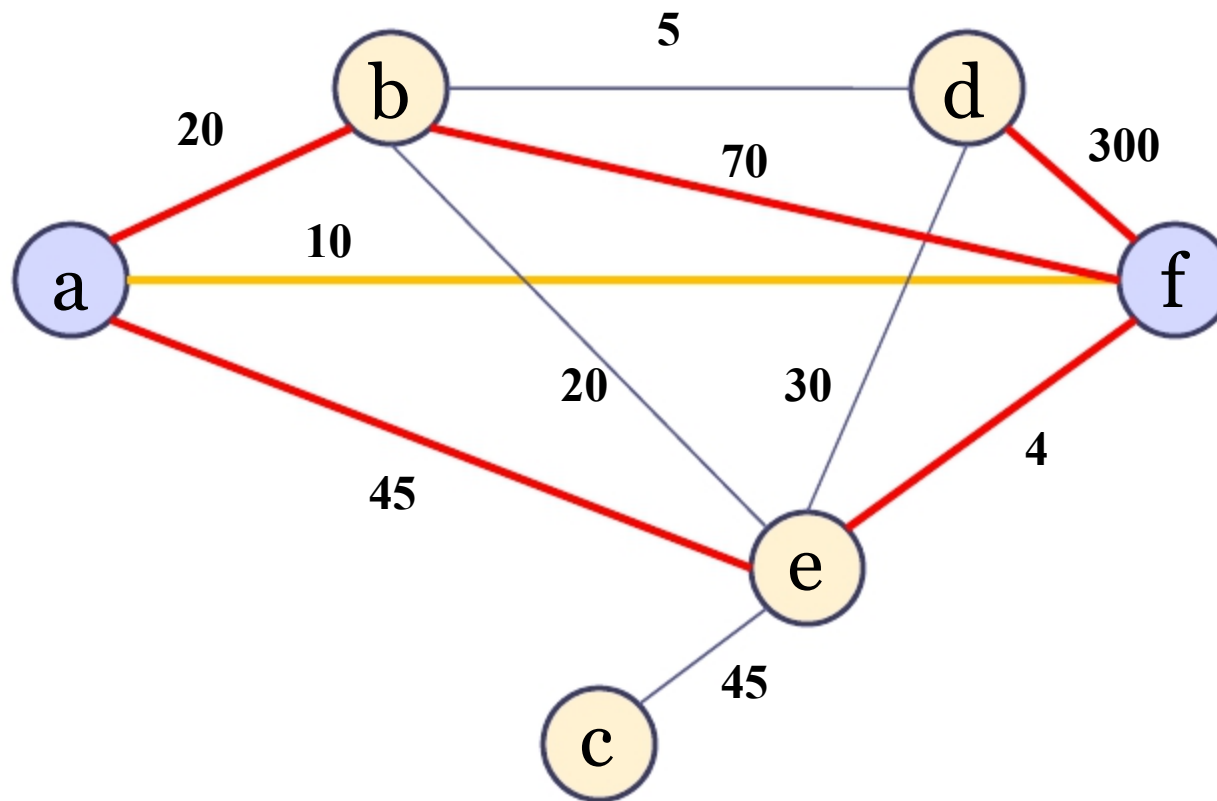


Conjunto S: {a,f}

Conjunto V-S: {b,c,d,e}

$X = \{(a,f)\}$

Aplique PRIM no seguinte grafo

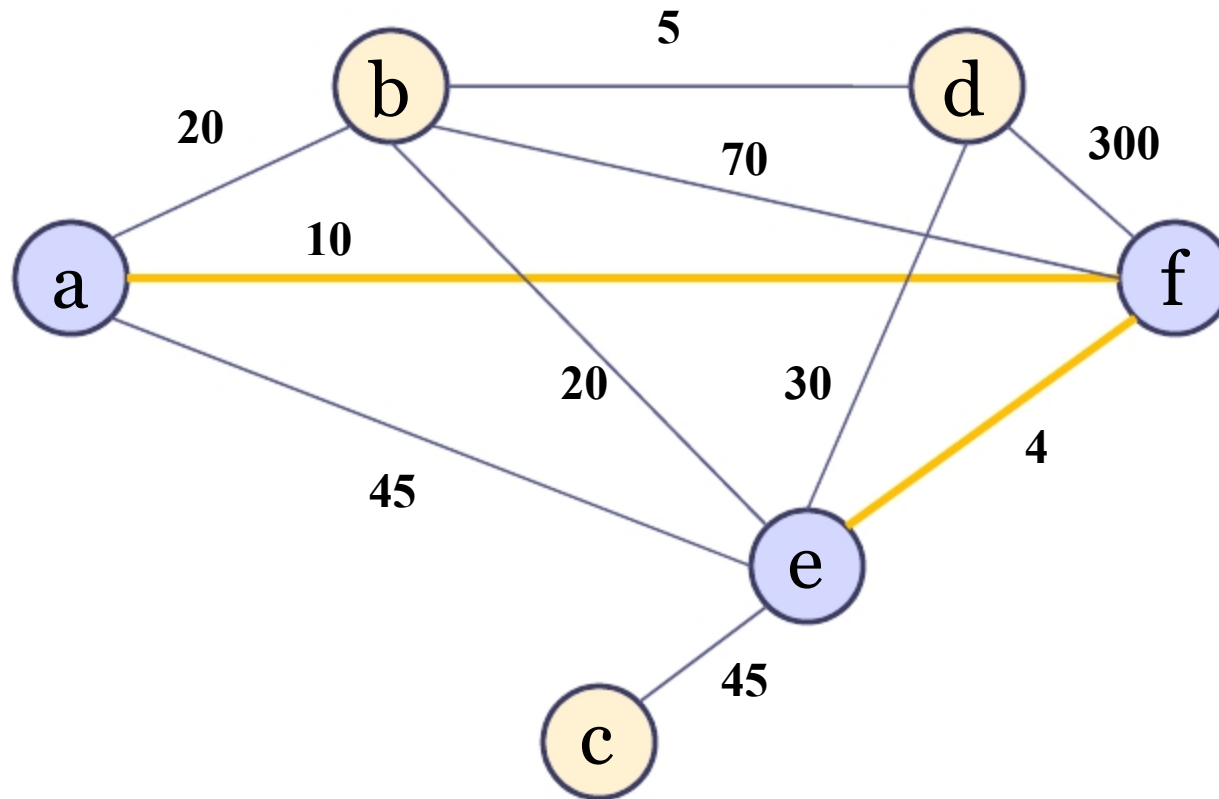


Conjunto S: {a,f}

Conjunto V-S: {b,c,d,e}

$X = \{(a,f)\}$

Aplique PRIM no seguinte grafo

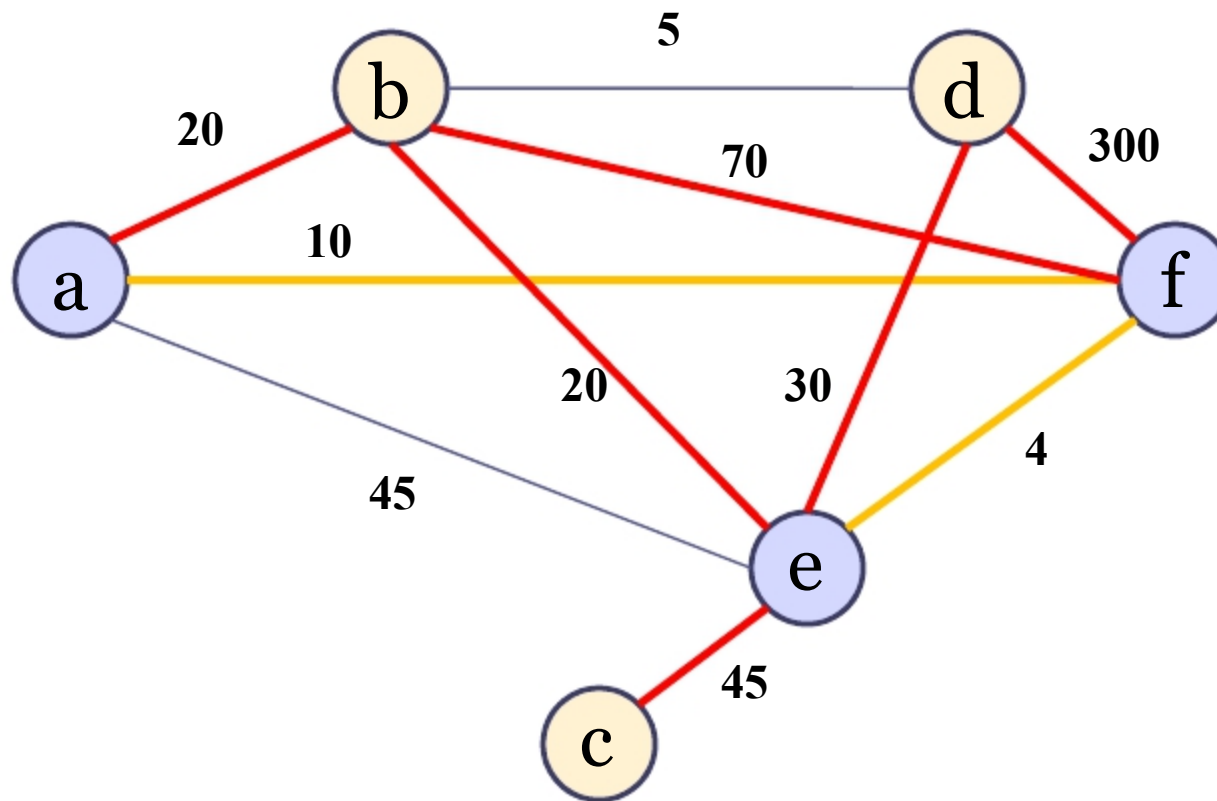


Conjunto S: {a,f,e}

Conjunto V-S: {b,c,d}

$X = \{(a,f), (f,e)\}$

Aplique PRIM no seguinte grafo

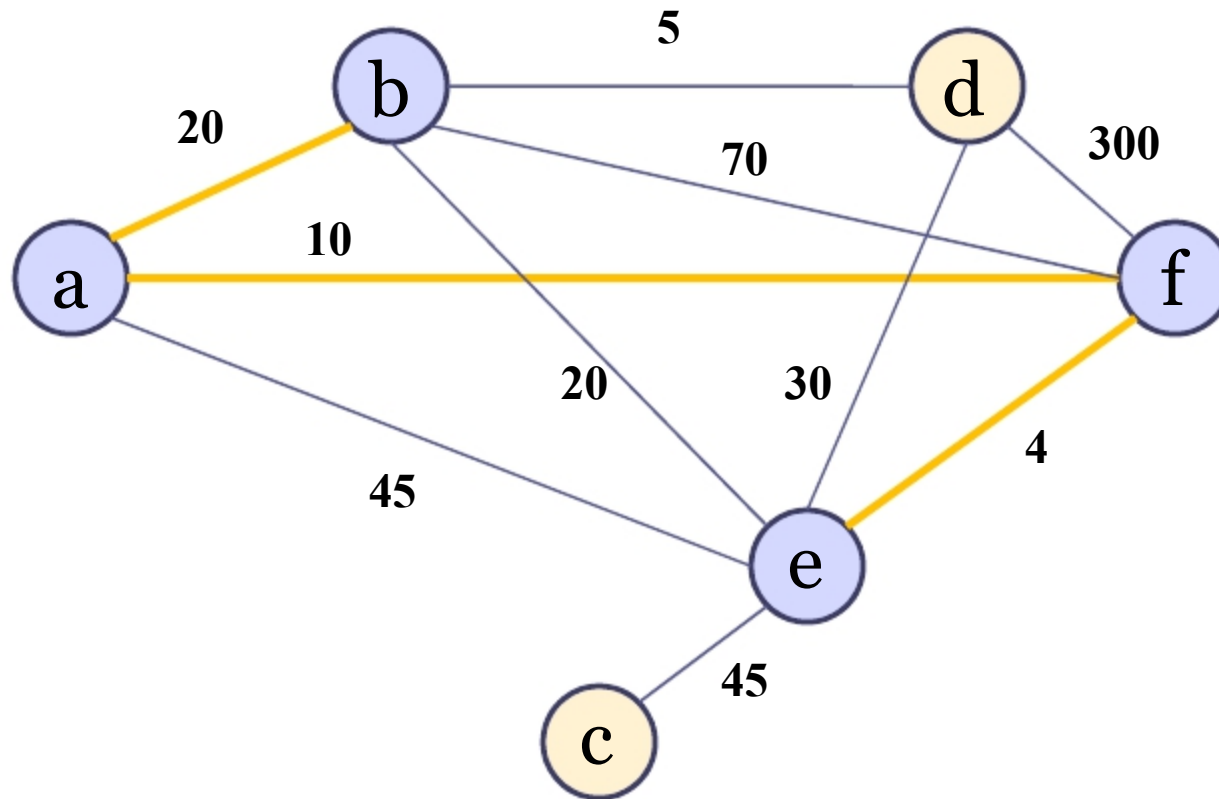


Conjunto S: {a,f,e}

Conjunto V-S: {b,c,d}

$X = \{(a,f), (f,e)\}$

Aplique PRIM no seguinte grafo

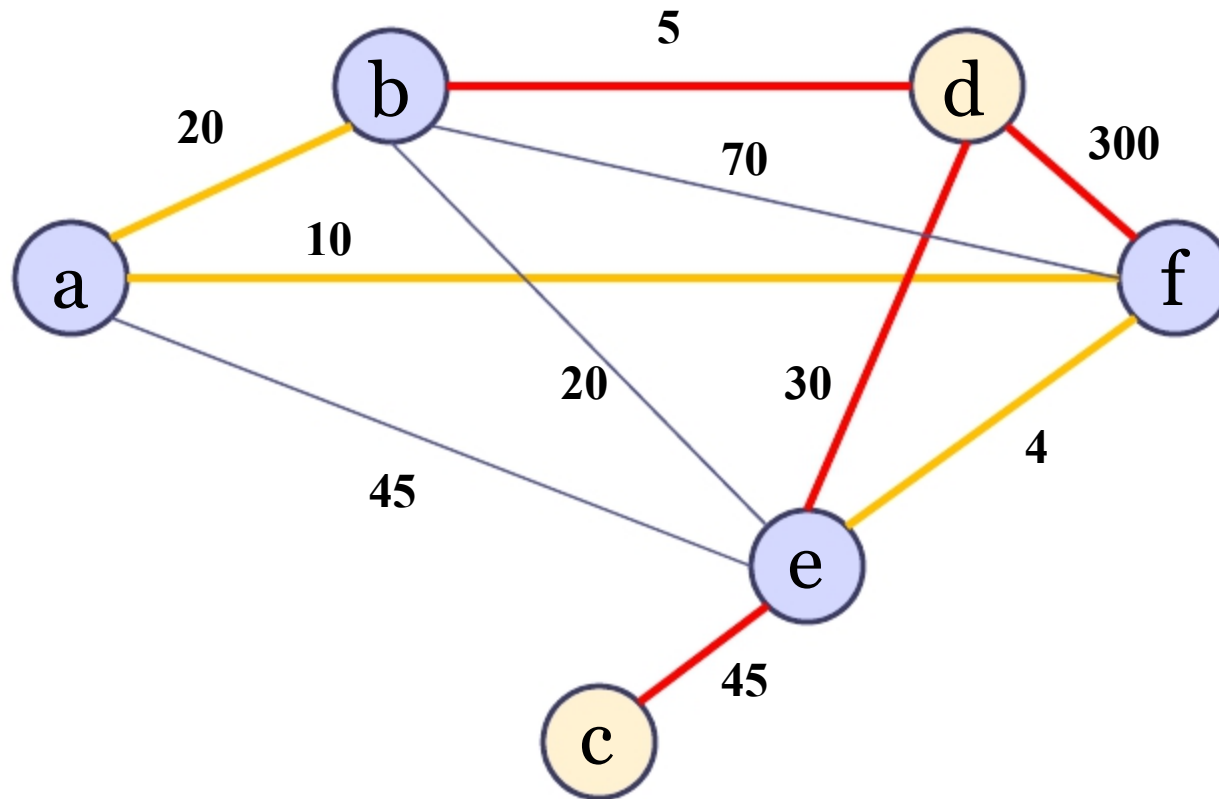


Conjunto S: {a,f,e,b}

Conjunto V-S: {c,d}

$X = \{(a,f), (f,e), (a,b)\}$

Aplique PRIM no seguinte grafo

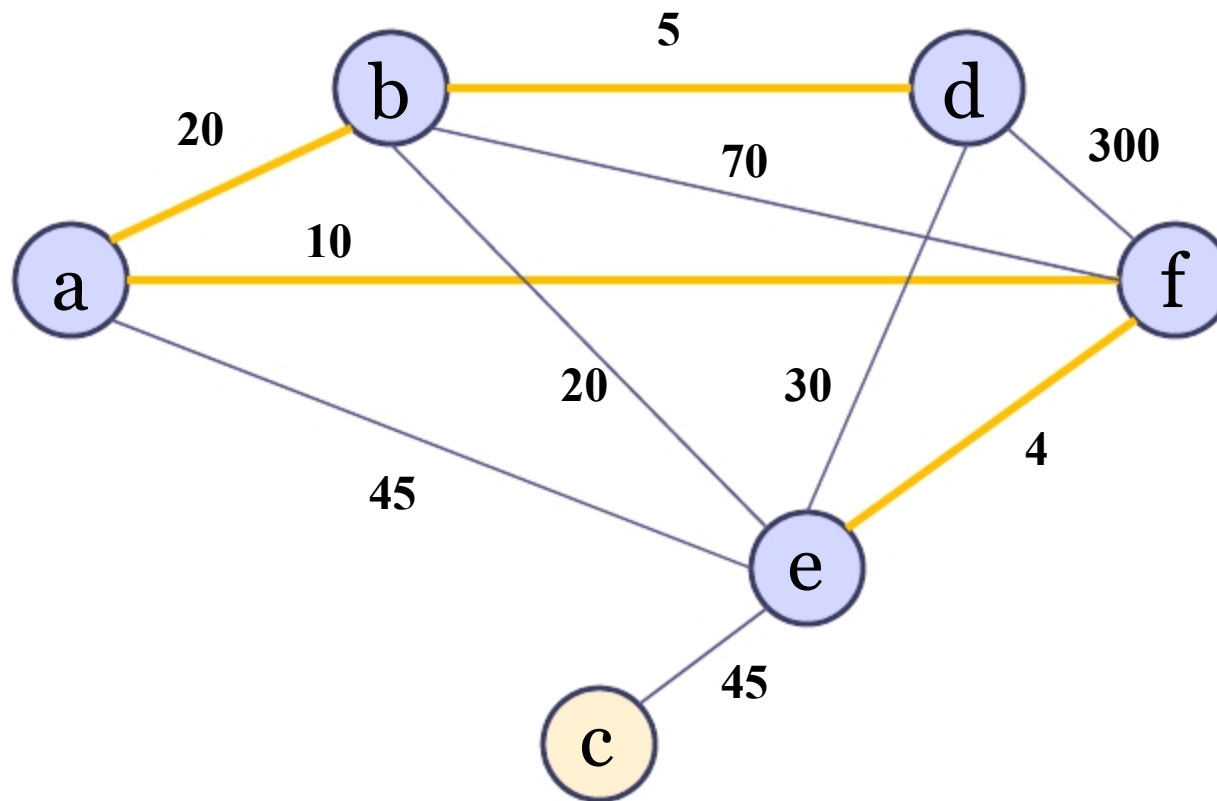


Conjunto S: {a,f,e,b}

Conjunto V-S: {c,d}

$X = \{(a,f), (f,e), (a,b)\}$

Aplique PRIM no seguinte grafo

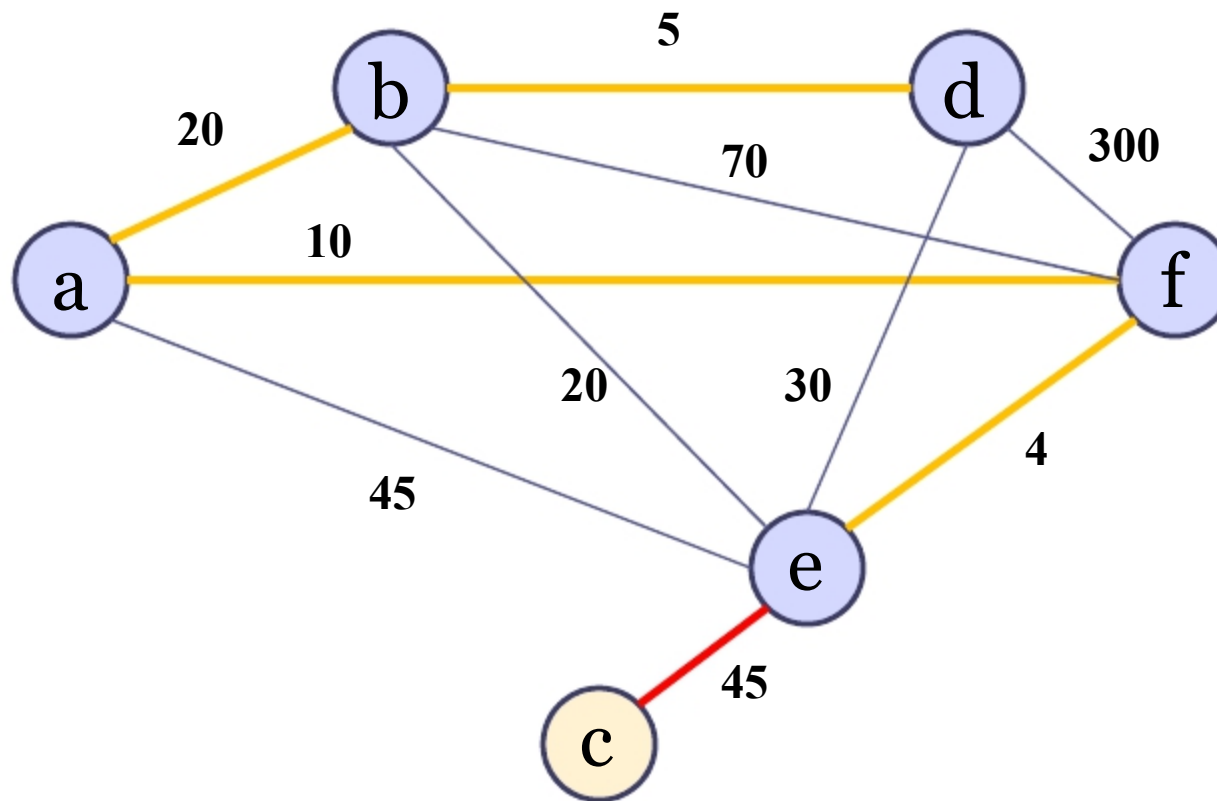


Conjunto S: {a,f,e,b,d}

Conjunto V-S: {c}

$X = \{(a,f), (f,e), (a,b), (b,d)\}$

Aplique PRIM no seguinte grafo

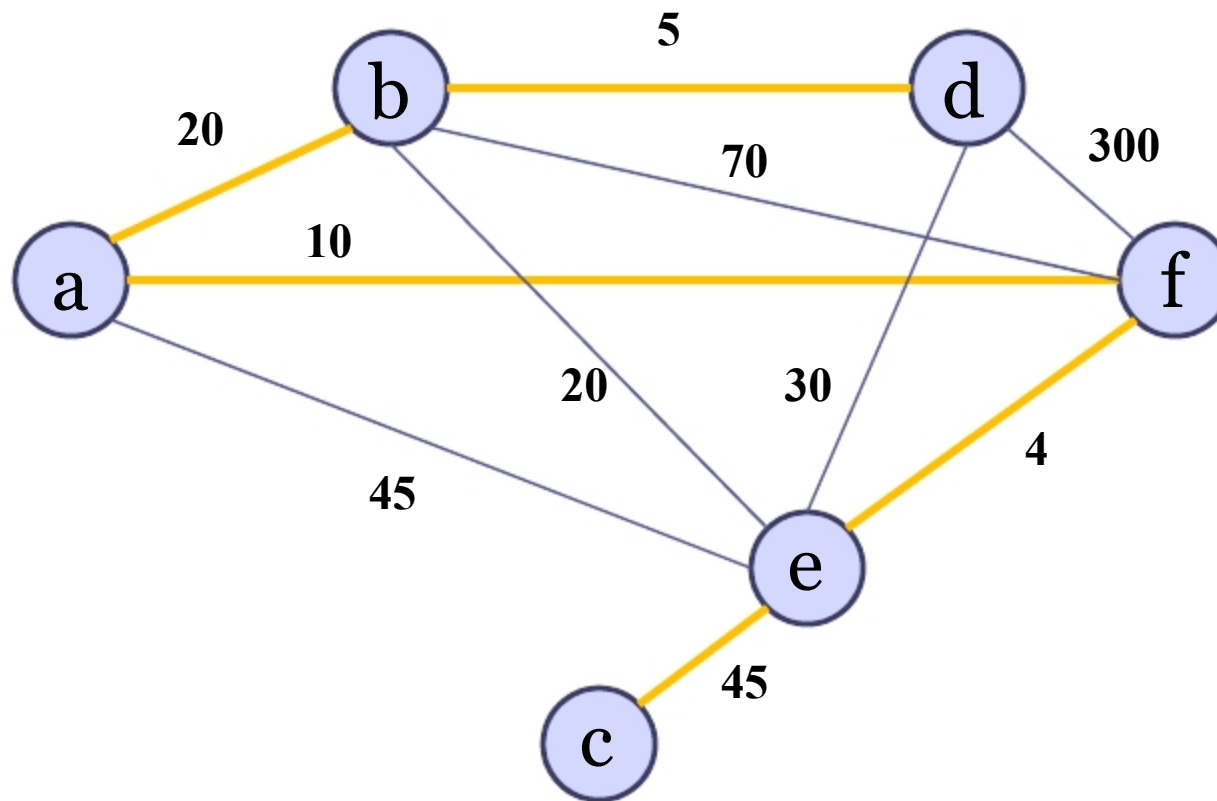


Conjunto S: {a,f,e,b,d}

Conjunto V-S: {c}

$X = \{(a,f), (f,e), (a,b), (b,d)\}$

Aplique PRIM no seguinte grafo



Conjunto S: {a,f,e,b,d,c}

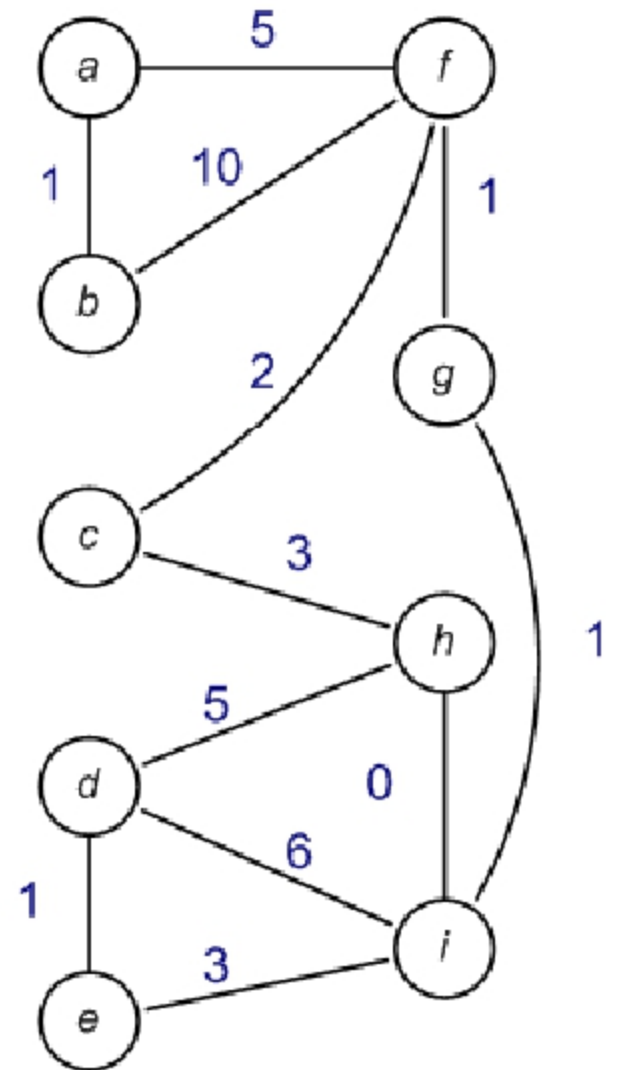
Conjunto V-S: { }

$X = \{(a,f), (f,e), (a,b), (b,d), (c,e)\}$

Custo da AGM: 84

Exercício

- AGM_PRIM((V,A), w, a)
 - Começando do vértice i ;



Curiosidades

- O algoritmo de *Prim* foi criado em 1930 por *Vojtech Jarnik* (matemático checo). Apenas 3 anos depois da proposta do problema;
- Em 1957, Prim reinventa o algoritmo quando trabalhava nos laboratórios Bell, ao lado de Kruskall (publicando o trabalho no *Bell Systems Technical Journal*);
- Em 1959 *Dijkstra* (matemático holandês) reinventa novamente o mesmo algoritmo;
- As vezes, na literatura podemos encontrar a referencia de *Algoritmo de Jarnik* para referenciar o mais conhecido popularmente como Algoritmo de Prim.



Bibliografia

- CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; (2002). Algoritmos – Teoria e Prática. Tradução da 2ª edição americana. Rio de Janeiro. Editora Campus.
- ZIVIANI, N. (2007). Projeto e Algoritmos com implementações em Java e C++. São Paulo. Editora Thomson;

