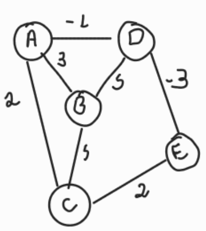
1) Os algoritmos de Prim e Kruskall são utilizados quando queremos encontrar uma árvore geradora resultante de um grafo G. Uma árvore geradora mínima é um subgrafo conexo T tal que $|V^T| = |V^G|$ e $|E^T| = |V^G| - 1$, de forma que a remoção de uma aresta de T o desconecta e a soma dos pesos das arestas de T é mínima, ou seja, não existe outra árvore geradora com a soma dos pesos das arestas menor que T.

Como o funcionamento dos algoritmos implica em minimizar a soma priorizando as arestas de menor peso, um grafo com peso positivo ou negativo não influenciará no resultado correto para geração de árvores mínimas em G.

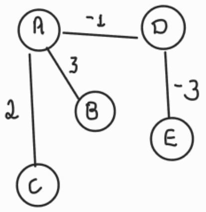
G



Prim iniciando com 1 vértice, procurará pela menor aresta conectada a v a adicionando no conjunto de arestas de T e em seguida procurará pela menor aresta conectada aos vértices então conhecidos sucessivamente até $|V^T| = |V^G|$.

se não fechar ciclo.

Para G começando em A



soma = (-1) + (-3) + 3 + 2
= 1

Kruskall funciona de forma a ordenar as arestas de forma não decrescente e iterar sob as arestas ordenadas enquanto $|E^T| < (|V| - 1)$ de forma que se a aresta não formar ciclo ela será adicionada à T.

Para G:

ordenação

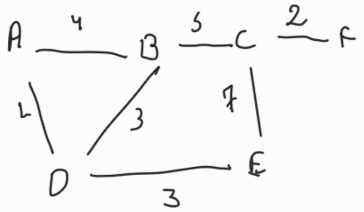
~~(D,E)~~ -3, ~~(A,D)~~ -1, ~~(A,C)~~ 2,
~~(C,E)~~ 2, ~~(A,B)~~ 3, (B,C) 5, (B,D) 5

$\lvert E^T \rvert$	$\lvert E^T \rvert < (V-1)$	vértices conhecidos	fecha ciclo	peso	soma
1	T	{D,E}	Não	-3	-3
2	T	{D,E,A}	Não	-1	-4
3	T	{D,E,A,C}	Não	2	-2
3	T	{D,E,A,C}	Sim	—	—
4	T	{D,E,A,C,B}	Não	3	1
5	F	—	—	—	1

2) Uma árvore geradora máxima

é um subgrafo conexo T tal que $|V^T| = |V^G|$ e $|E^T| = |V^G| - 1$, de forma que a remoção de uma aresta de T o desconecta e a soma dos pesos das arestas de T é maxima, ou seja, não existe outra árvore geradora com a soma dos pesos das arestas maior que T.

G exemplo



Krushall :

1) Ordenar as arestas de G por peso de forma não crescente.

ordem:

~~(C,E)~~ 7, ~~(B,C)~~ 5, ~~(A,B)~~ 4
~~(B,D)~~ 3, ~~(D,E)~~ 3, ~~(C,F)~~ 2, ~~(A,D)~~ 1

2) Enquanto $|E^T| < (|V| - 1)$

3) Selecionar aresta de maior peso

4) Se aresta selecionada não formar ciclo:

Um ciclo será formado se os vértices de origem e destino fizerem parte do mesmo subconjunto de vértices conhecidos.

5) Adicionar aresta selecionada a $|E^T|$.

6) remover aresta selecionada da ordenação

Passo a passo

Aresta	$\vert E^T\vert < (V-1)$	$\vert E^T\vert$	vértices conhecidos	fecha ciclo	peso	soma
(C,E)	T	1	{ }	N	7	7
(B,C)	T	2	{C,E}	N	5	12
(A,B)	T	3	{C,E,B}	N	4	16
(B,D)	T	4	{C,E,B,A}	N	3	19
(D,E)	T	4	{C,E,B,A,D}	S	3	—
(C,F)	T	5	{C,E,B,A,D}	N	2	21
(A,D)	T	5	{C,E,B,A,D}	S	—	21

$\boxed{\text{Soma} = 21}$

Prim

1) Escolher um vértice $X \in V$ tal que $|V^r| = \{x\}$ e $|E^r| = 0$

2) Enquanto $|V^r| \neq |V|$

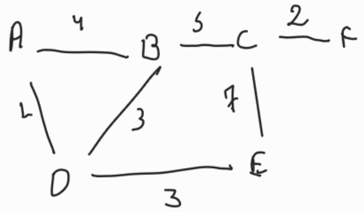
3) Selecionar a maior aresta (x, w) tal que $x \in V^r$ e $w \in V$

4) Se aresta (x, w) selecionada não fechar ciclo

5) Adicionar aresta (x, w) a $|E^r|$ e adicionar w a V^r.

vértices de origem e destino não fazem parte do mesmo subconjunto de vértices conhecidos.

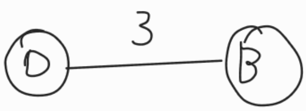
G)



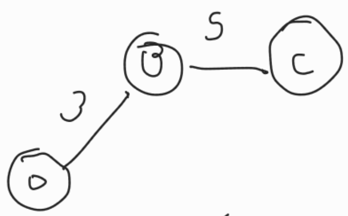
Passo a passo

1) 

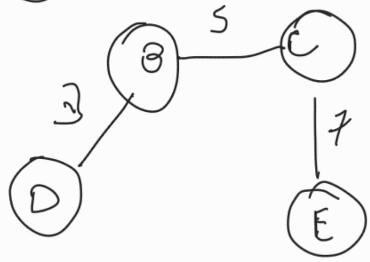
$V^r = 1$

2) 

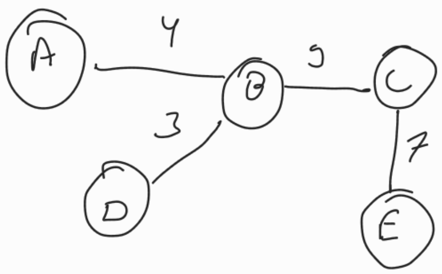
$V^r = 2$



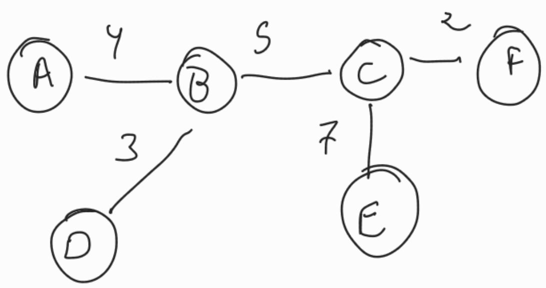
$V^r = 3$



$V^r = 4$



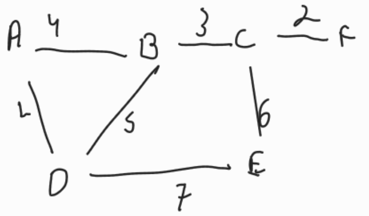
$V^r = 5$



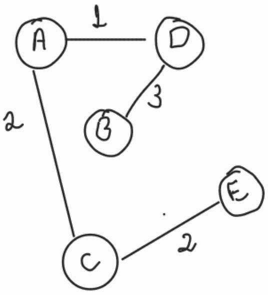
$V^r = 6$

$\boxed{\text{Soma} = 21}$

3) AGM → revisitar



4) G



1) $|E'| = |E|$

$|V'| = |V|$

2) ordenação:

~~(C,B)~~ 5, ~~(D,E)~~ 4, ~~(A,B)~~ 3,

(B,D) 3, (A,C) 2, (C,E) 2

(A,D) 1

3) Enquanto $|E'| > (|V| - 1)$

4) Selecionar maior aresta ordenada

5) Se aresta pechar ciclo

6) remover aresta selecionada de $|E'|$

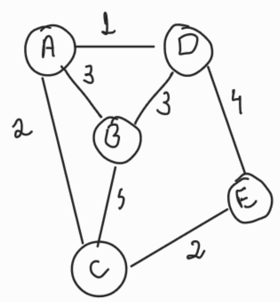
vértices de destino e origem fazem parte do mesmo subconjunto de vértices resultantes.

Execução

aresta	vértices conhecidos	fecha ciclo	soma	$\|E'\|$
~~(C,B)~~	{A,B,C,D,E}	S	15	6
~~(D,E)~~	{A,B,C,D,E}	S	11	5
~~(A,B)~~	{A,B,C,D,E}	S	8	4

$\boxed{\text{Soma} = 8}$

G



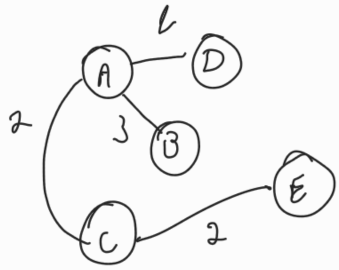
Kruskall normal:

~~(A,D)~~, ~~(A,C)~~, ~~(C,E)~~,
1, 2, 2

~~(A,B)~~, (B,D), (D,E)
3, 3, 4

(B,C)
5

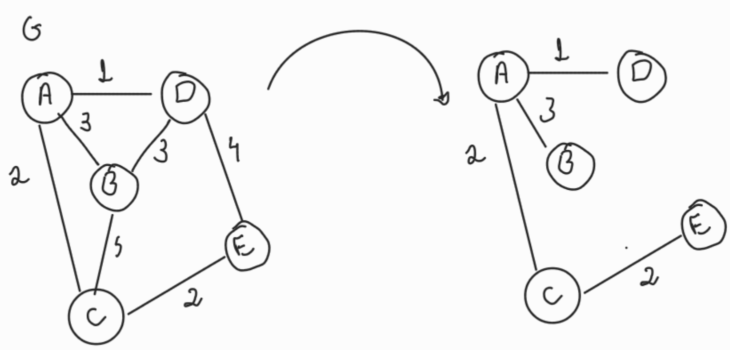
aresta	vértices conhecidos	fecha ciclo	soma	|E'|
(A,D)	{ }	N	1	1
(A,C)	{A,D}	N	3	2
(C,E)	{A,D,C}	N	5	3
(A,B)	{A,D,C,E}	N	8	4
			$\boxed{\text{soma} = 8}$	parar



Conclui-se que, sim, o algoritmo encontraria corretamente a árvore geradora mínima pois seleciona a maior aresta que está fechando o ciclo e então a remove.

5)

G T



A) w(e) = 15 → correto

Pesos

	1	2	2	3	3	4	5
tcs	16	17	17	18	18	19	20

→ Árvore continua a mesma pois o algoritmo seleciona a menor aresta sucessivamente

B) w(e) x 15

	1	2	2	3	3	4	5
x15	15	30	30	45	45	60	75

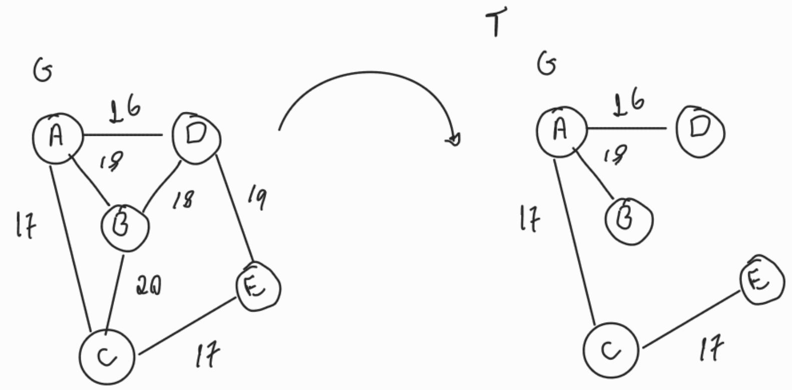
→ Árvore continua a mesma pois o algoritmo seleciona a menor aresta sucessivamente

C) w(e) x w(e)

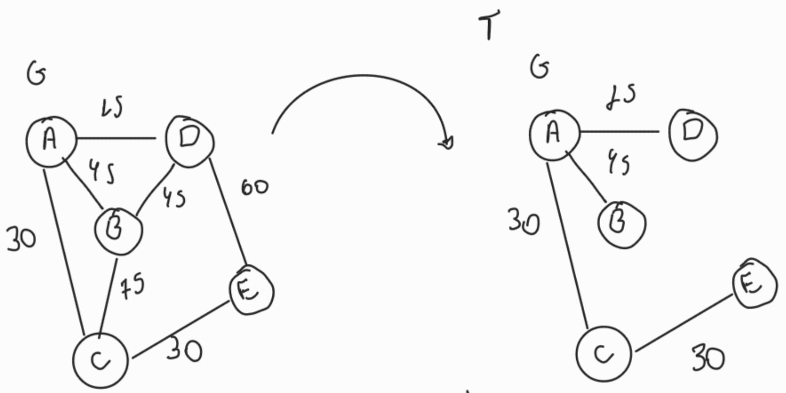
	1	2	2	3	3	4	5
x w(e)	1	4	4	9	9	16	25

→ árvore continua a mesma pois o algoritmo seleciona a menor aresta sucessivamente.

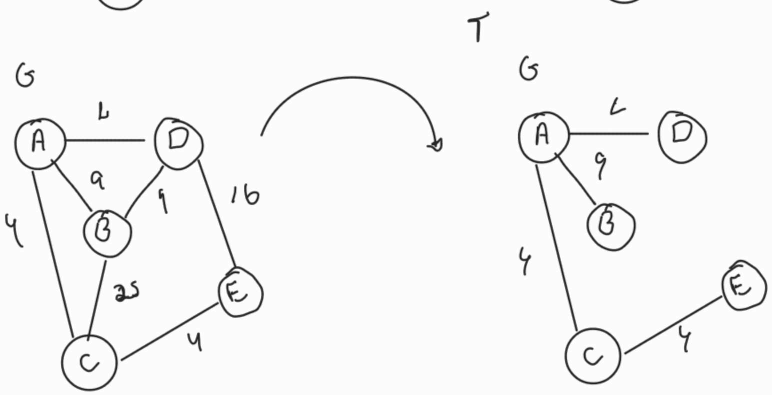
A)



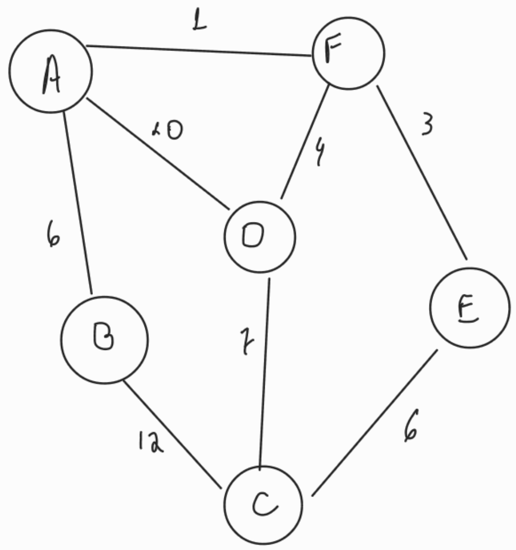
B)

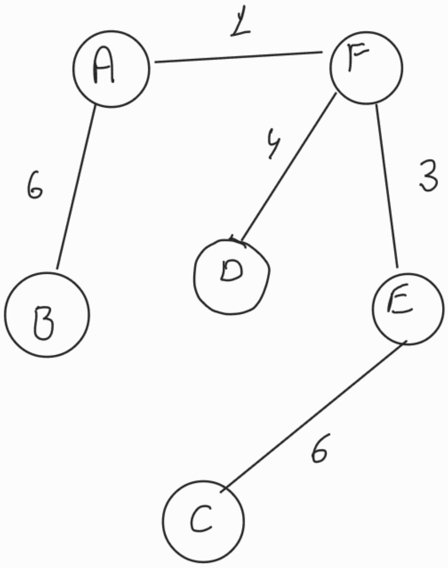


C)



6)





maior peso atual = 6

remove :