

Torres de Hanoi

- É um exemplo clássico da aplicabilidade da recursão.
- Deseja-se mover n discos, um de cada vez, de uma torre de origem para outra de destino, usando uma terceira auxiliar, sem nunca colocar um disco maior sobre outro menor.
- Exemplo para 3 discos:



Torres de Hanoi

- É um exemplo clássico da aplicabilidade da recursão.
- Deseja-se mover n discos, um de cada vez, de uma torre de origem para outra de destino, usando uma terceira auxiliar, sem nunca colocar um disco maior sobre outro menor.
- Exemplo para 3 discos:



Torres de Hanoi

- É um exemplo clássico da aplicabilidade da recursão.
- Deseja-se mover n discos, um de cada vez, de uma torre de origem para outra de destino, usando uma terceira auxiliar, sem nunca colocar um disco maior sobre outro menor.
- Exemplo para 3 discos:



Torres de Hanoi

- É um exemplo clássico da aplicabilidade da recursão.
- Deseja-se mover n discos, um de cada vez, de uma torre de origem para outra de destino, usando uma terceira auxiliar, sem nunca colocar um disco maior sobre outro menor.
- Exemplo para 3 discos:



Torres de Hanoi

- É um exemplo clássico da aplicabilidade da recursão.
- Deseja-se mover n discos, um de cada vez, de uma torre de origem para outra de destino, usando uma terceira auxiliar, sem nunca colocar um disco maior sobre outro menor.
- Exemplo para 3 discos:



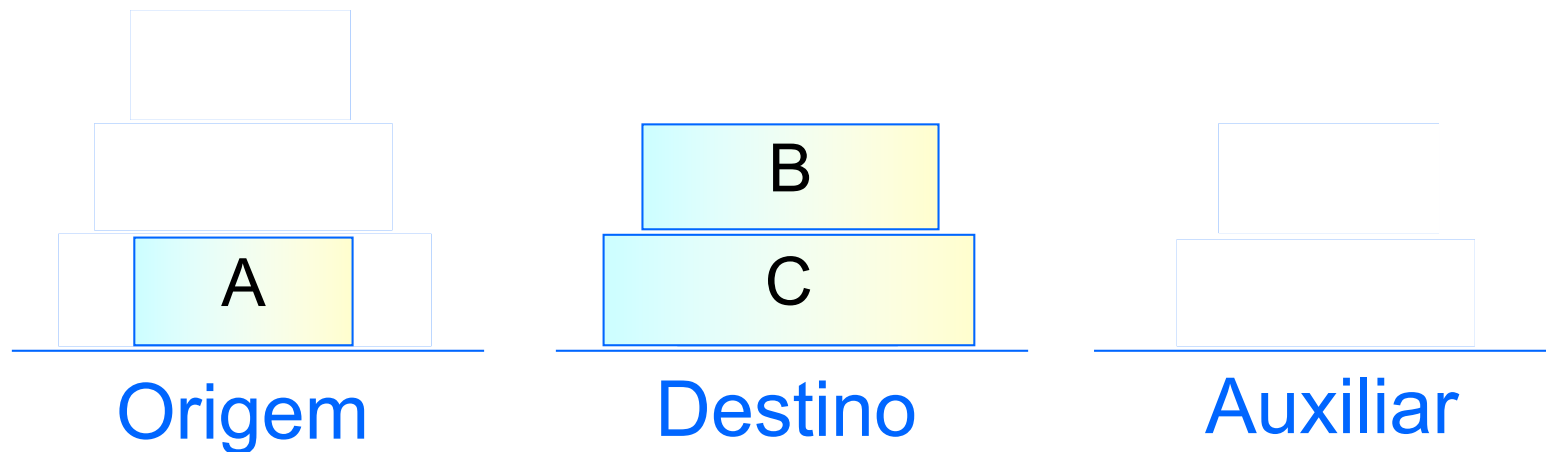
Torres de Hanoi

- É um exemplo clássico da aplicabilidade da recursão.
- Deseja-se mover n discos, um de cada vez, de uma torre de origem para outra de destino, usando uma terceira auxiliar, sem nunca colocar um disco maior sobre outro menor.
- Exemplo para 3 discos:



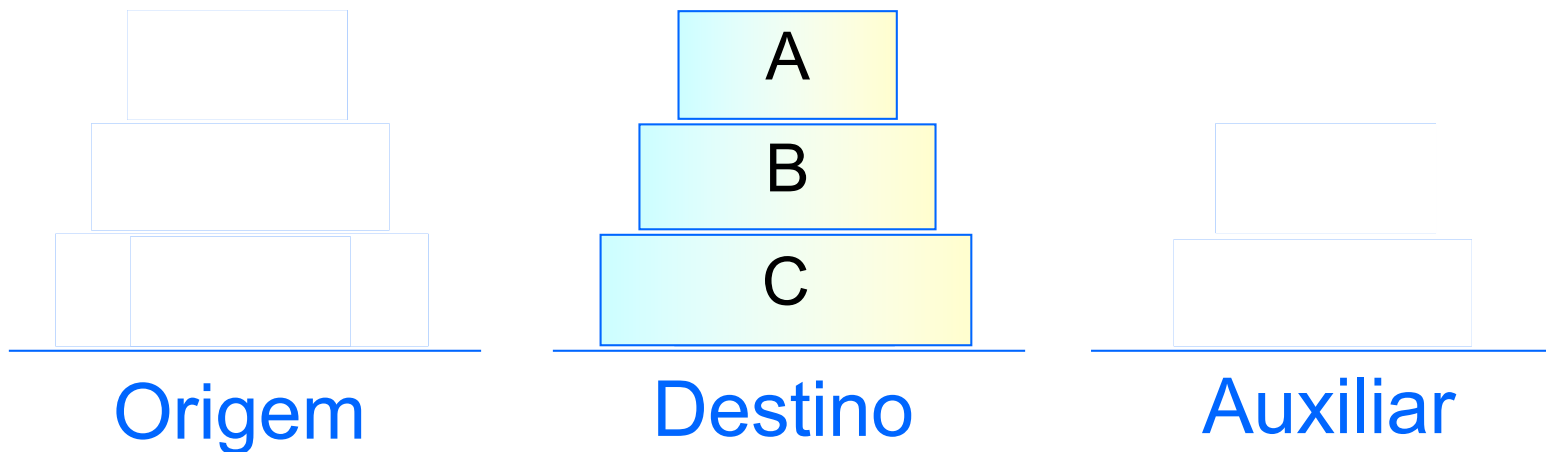
Torres de Hanoi

- É um exemplo clássico da aplicabilidade da recursão.
- Deseja-se mover n discos, um de cada vez, de uma torre de origem para outra de destino, usando uma terceira auxiliar, sem nunca colocar um disco maior sobre outro menor.
- Exemplo para 3 discos:



Torres de Hanoi

- É um exemplo clássico da aplicabilidade da recursão.
- Deseja-se mover n discos, um de cada vez, de uma torre de origem para outra de destino, usando uma terceira auxiliar, sem nunca colocar um disco maior sobre outro menor.
- Exemplo para 3 discos:



Remoção da raiz de um *heap*



Remoção da raiz de um *heap*



- Remover a raiz equivale a extrair o máximo valor presente no *heap*.

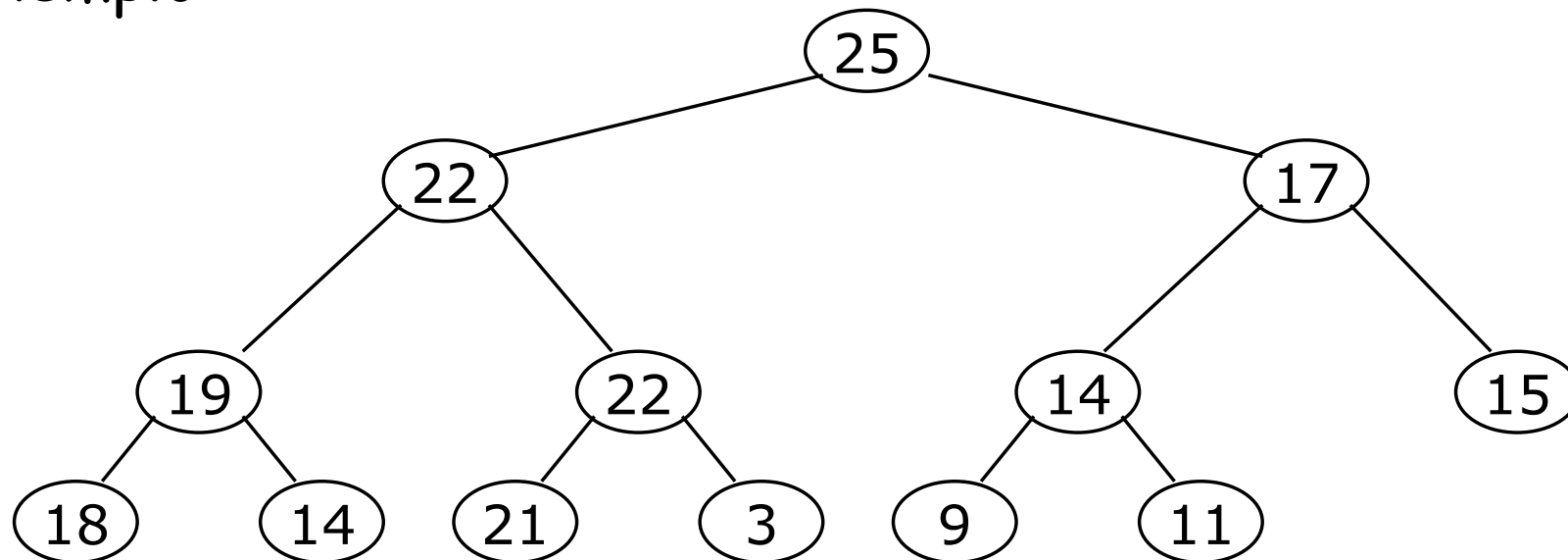
Remoção da raiz de um *heap*



- Remover a raiz equivale a extrair o máximo valor presente no *heap*.
- Em seguida, colocaremos em seu lugar a última folha e recuperaremos a propriedade estrutural com a aplicação de vários *Sifts*.

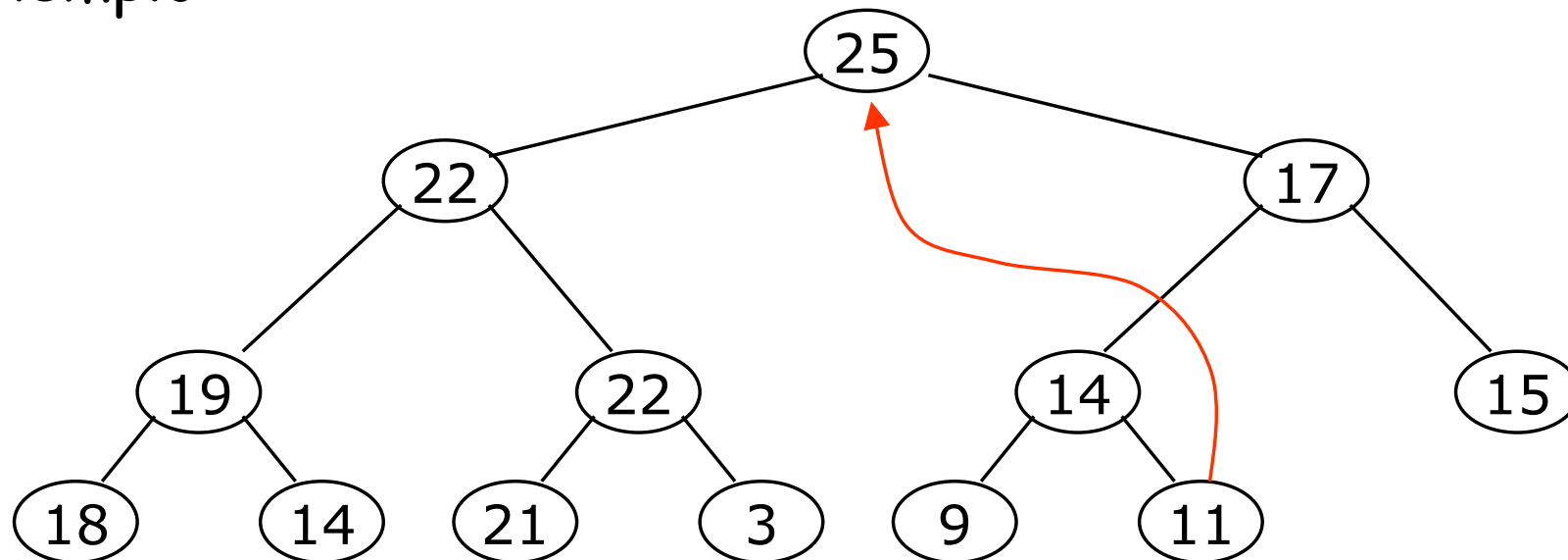
Remoção da raiz de um *heap*

- Remover a raiz equivale a extrair o máximo valor presente no *heap*.
- Em seguida, colocaremos em seu lugar a última folha e recuperaremos a propriedade estrutural com a aplicação de vários *Sifts*.
- Exemplo:



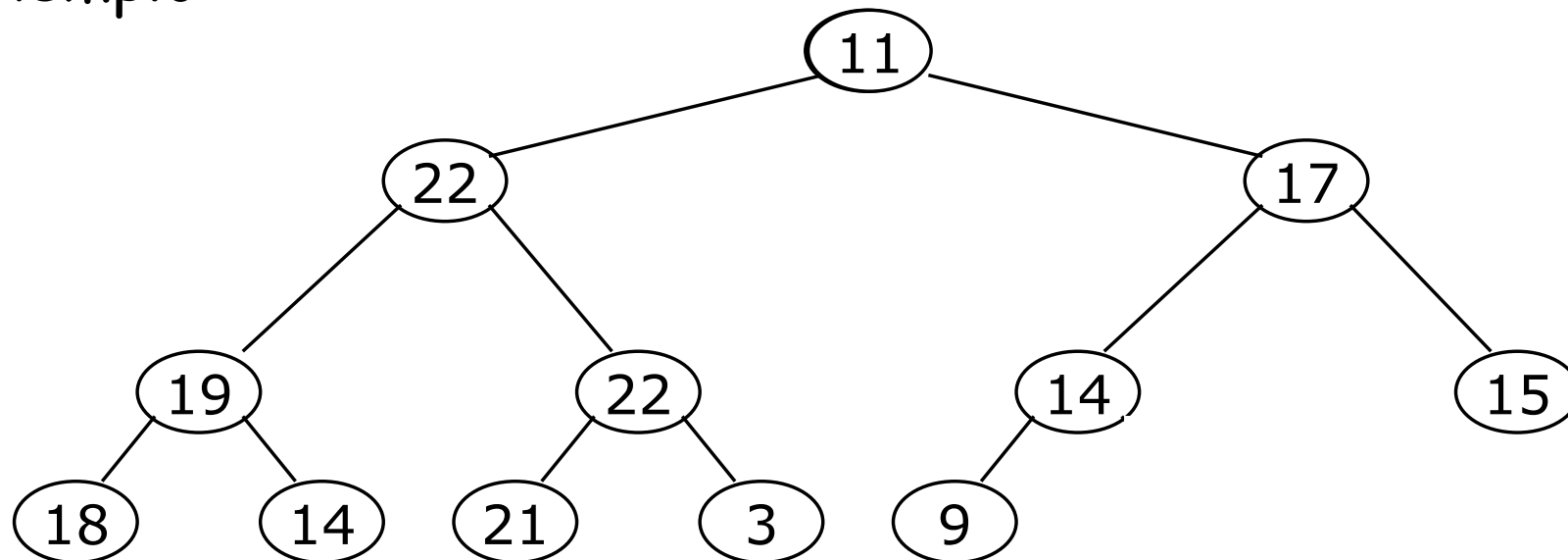
Remoção da raiz de um *heap*

- Remover a raiz equivale a extrair o máximo valor presente no *heap*.
- Em seguida, colocaremos em seu lugar a última folha e recuperaremos a propriedade estrutural com a aplicação de vários *Sifts*.
- Exemplo:



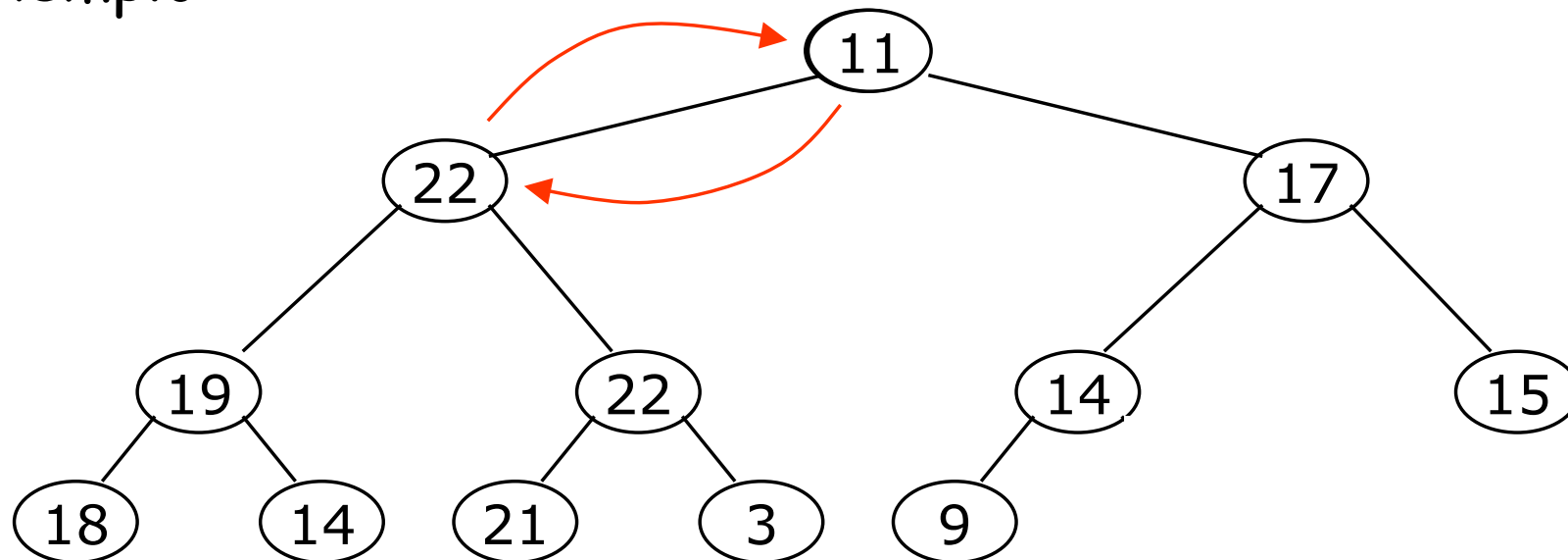
Remoção da raiz de um *heap*

- Remover a raiz equivale a extrair o máximo valor presente no *heap*.
- Em seguida, colocaremos em seu lugar a última folha e recuperaremos a propriedade estrutural com a aplicação de vários *Sifts*.
- Exemplo:



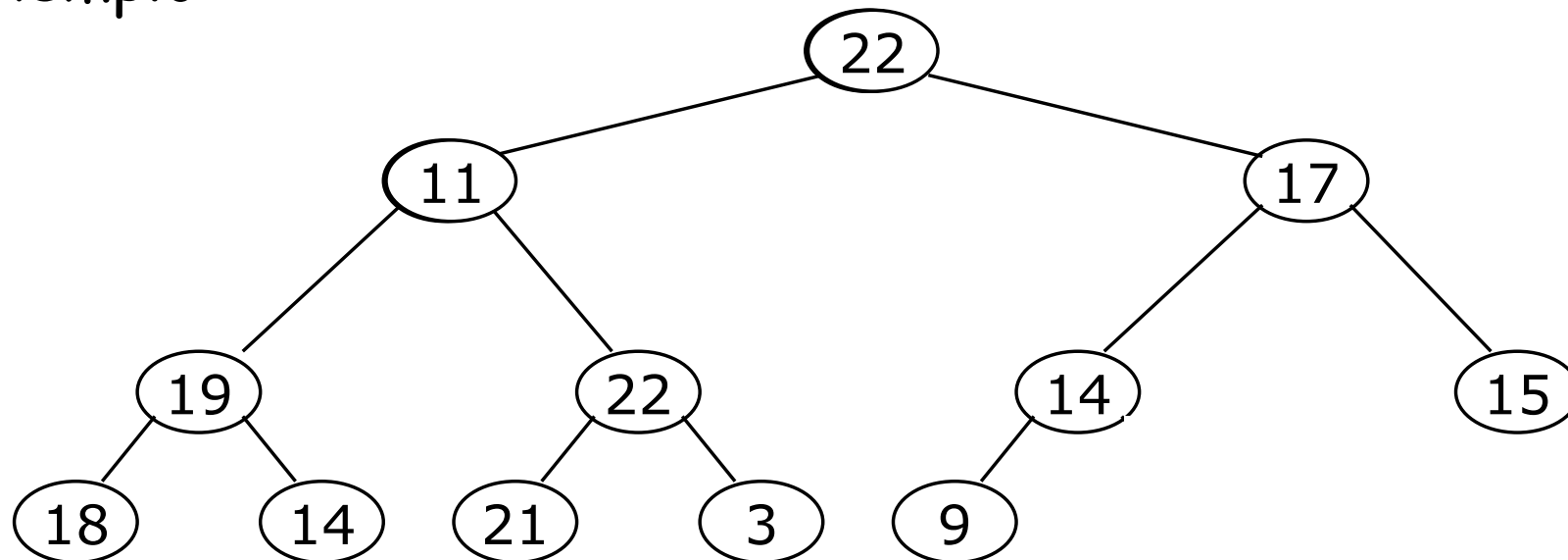
Remoção da raiz de um *heap*

- Remover a raiz equivale a extrair o máximo valor presente no *heap*.
- Em seguida, colocaremos em seu lugar a última folha e recuperaremos a propriedade estrutural com a aplicação de vários *Sifts*.
- Exemplo:



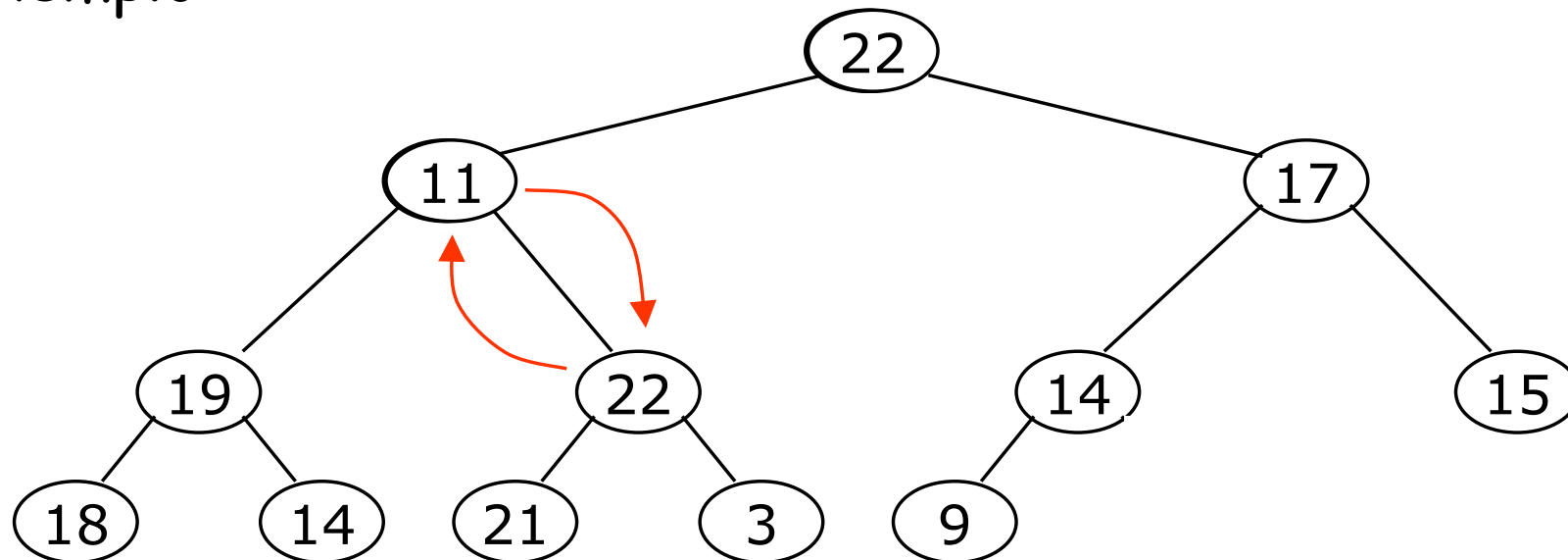
Remoção da raiz de um *heap*

- Remover a raiz equivale a extrair o máximo valor presente no *heap*.
- Em seguida, colocaremos em seu lugar a última folha e recuperaremos a propriedade estrutural com a aplicação de vários *Sifts*.
- Exemplo:



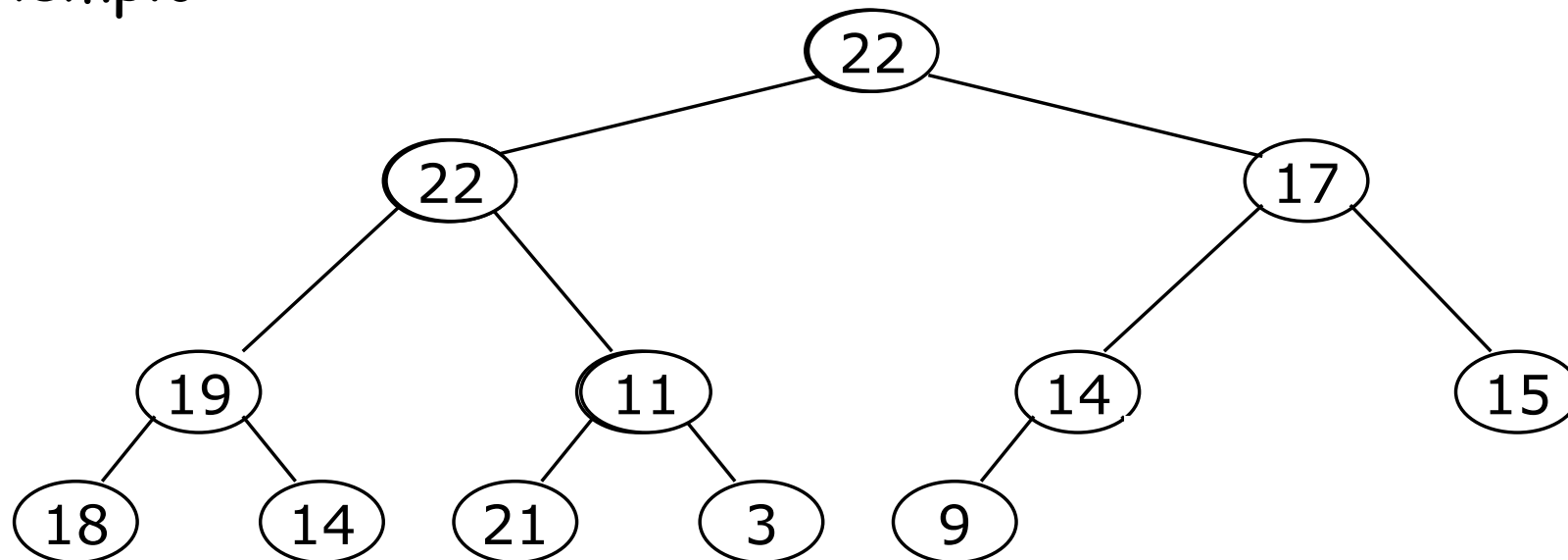
Remoção da raiz de um *heap*

- Remover a raiz equivale a extrair o máximo valor presente no *heap*.
- Em seguida, colocaremos em seu lugar a última folha e recuperaremos a propriedade estrutural com a aplicação de vários *Sifts*.
- Exemplo:



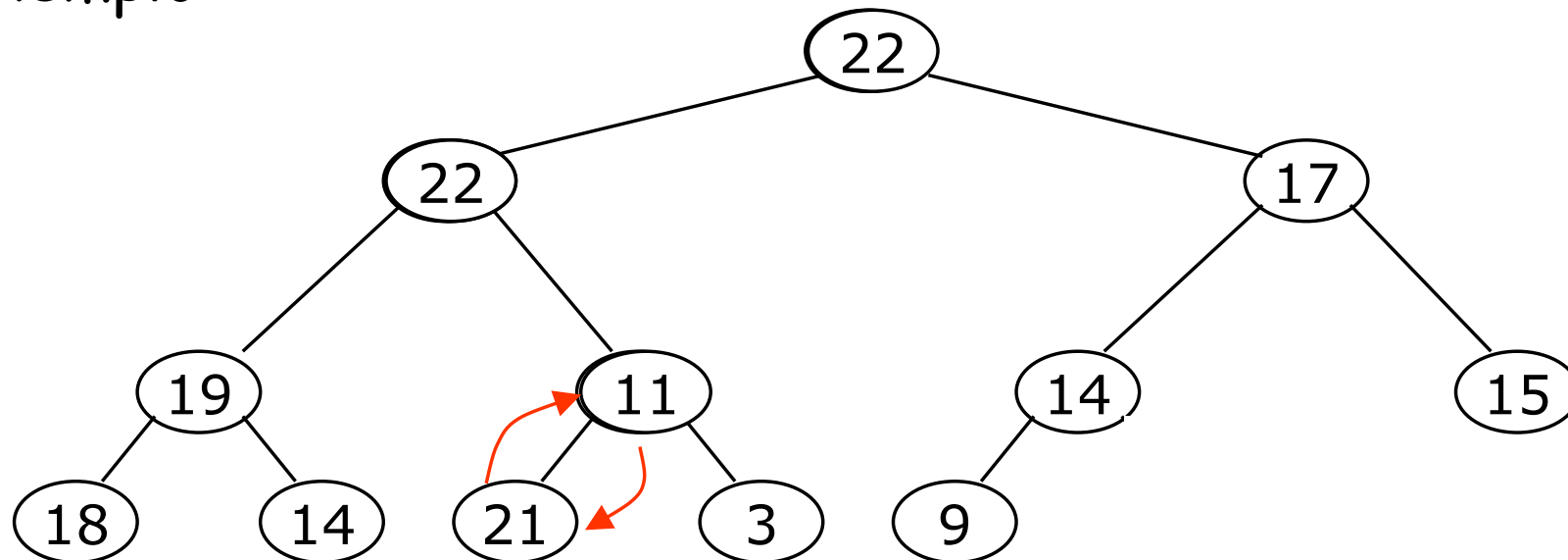
Remoção da raiz de um *heap*

- Remover a raiz equivale a extrair o máximo valor presente no *heap*.
- Em seguida, colocaremos em seu lugar a última folha e recuperaremos a propriedade estrutural com a aplicação de vários *Sifts*.
- Exemplo:



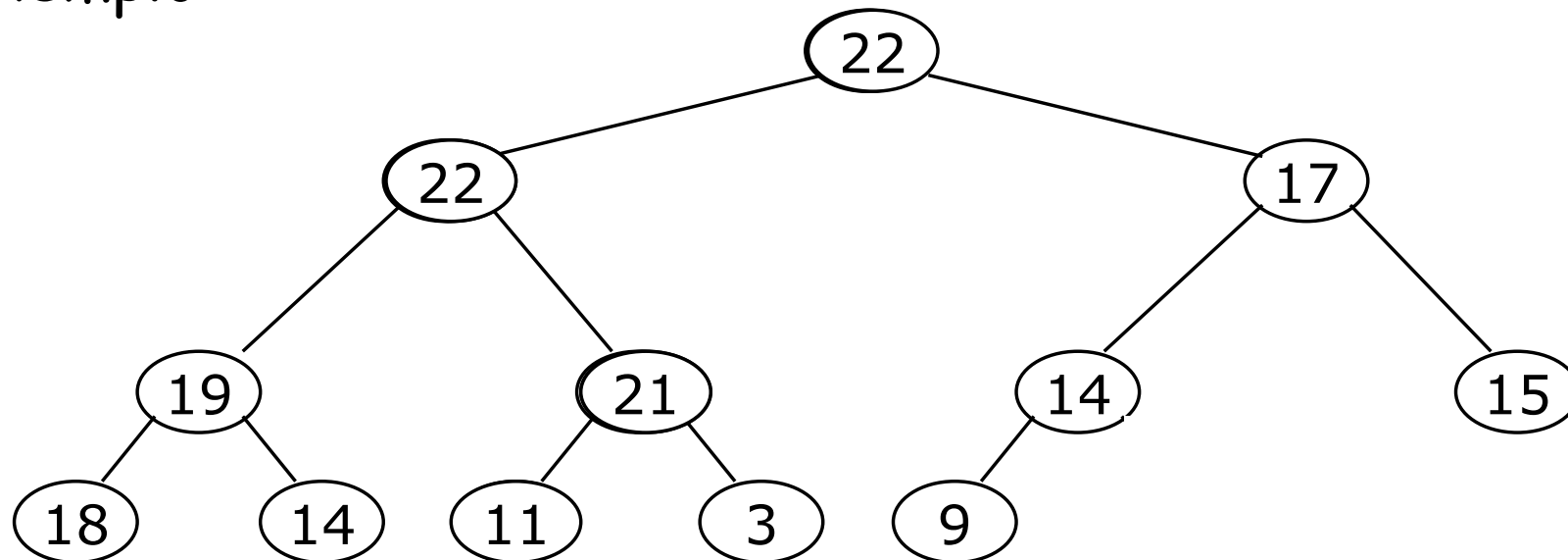
Remoção da raiz de um *heap*

- Remover a raiz equivale a extrair o máximo valor presente no *heap*.
- Em seguida, colocaremos em seu lugar a última folha e recuperaremos a propriedade estrutural com a aplicação de vários *Sifts*.
- Exemplo:



Remoção da raiz de um *heap*

- Remover a raiz equivale a extrair o máximo valor presente no *heap*.
- Em seguida, colocaremos em seu lugar a última folha e recuperaremos a propriedade estrutural com a aplicação de vários *Sifts*.
- Exemplo:




Exemplo: cálculo da função de falha

j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0									

(a) (b)

Exemplo: cálculo da função de falha

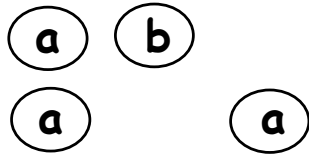


j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0									


Diagram illustrating the calculation of the failure function $F[j]$ for a string P . The string P is "abababababab". The failure function $F[j]$ is shown below the string, with values 0 for $j=0$ and $j=1$, and empty cells for $j=2$ through $j=10$. The diagram shows the characters 'a' and 'b' at positions 0 and 1, and 'a' at positions 2 and 3, illustrating the matching process for the failure function calculation.

Exemplo: cálculo da função de falha

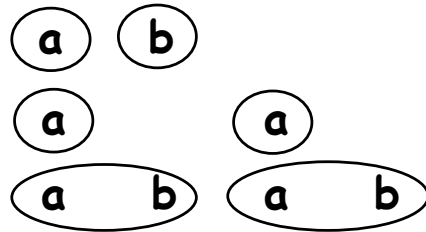
j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0	1								




Exemplo: cálculo da função de falha



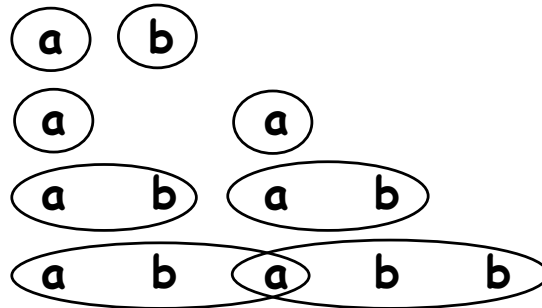
j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0	1								



Exemplo: cálculo da função de falha

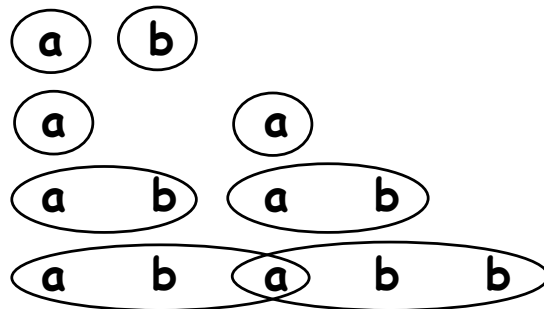


j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0	1	2							




Exemplo: cálculo da função de falha

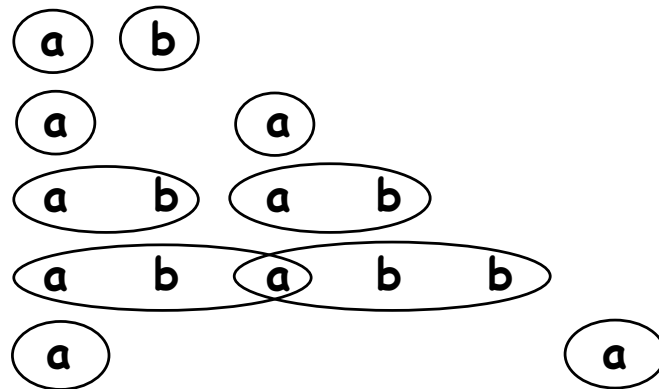
j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0	1	2	0						



Exemplo: cálculo da função de falha

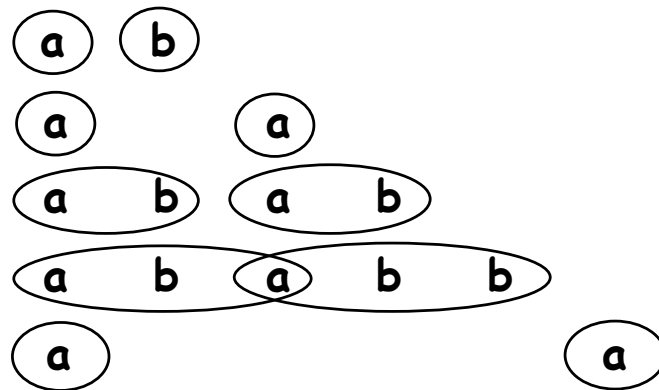


j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0	1	2	0						




Exemplo: cálculo da função de falha

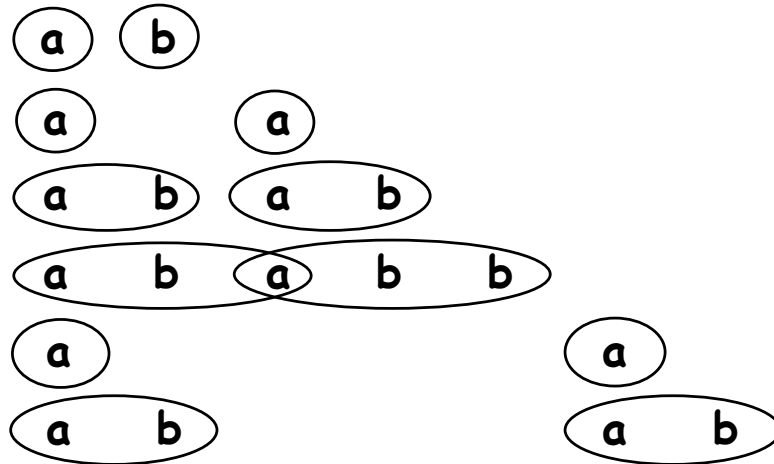
j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0	1	2	0	1					



Exemplo: cálculo da função de falha

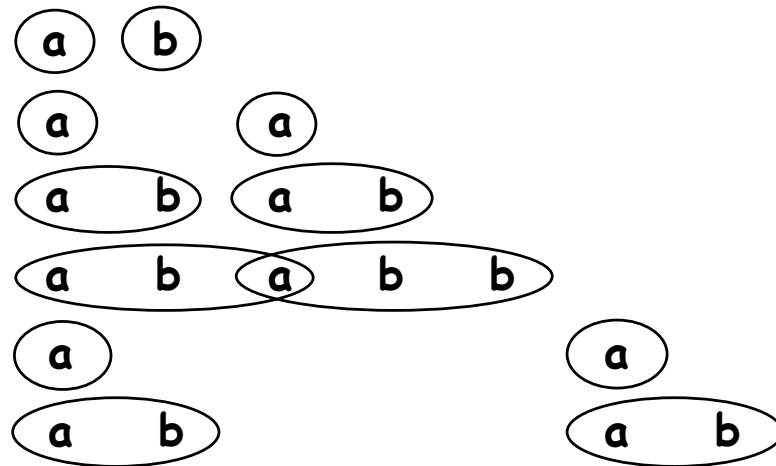


j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0	1	2	0	1					




Exemplo: cálculo da função de falha

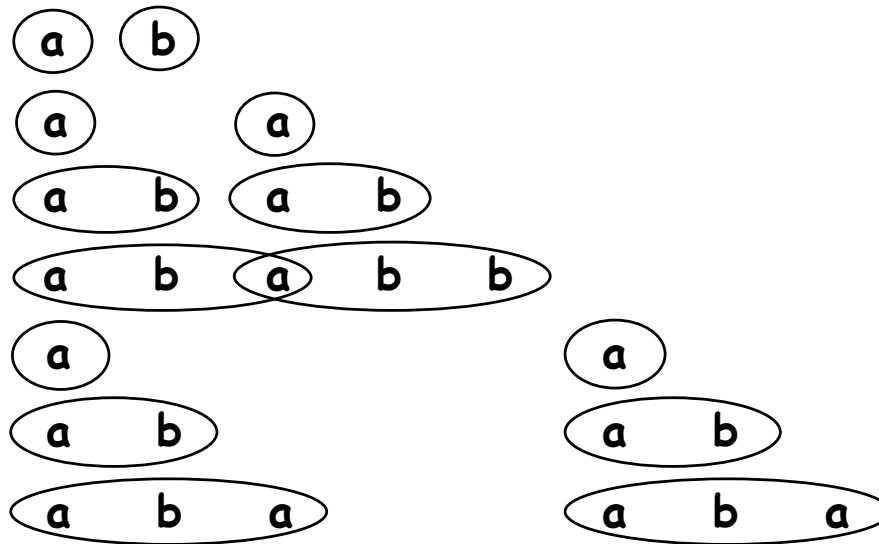
j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0	1	2	0	1	2				



Exemplo: cálculo da função de falha

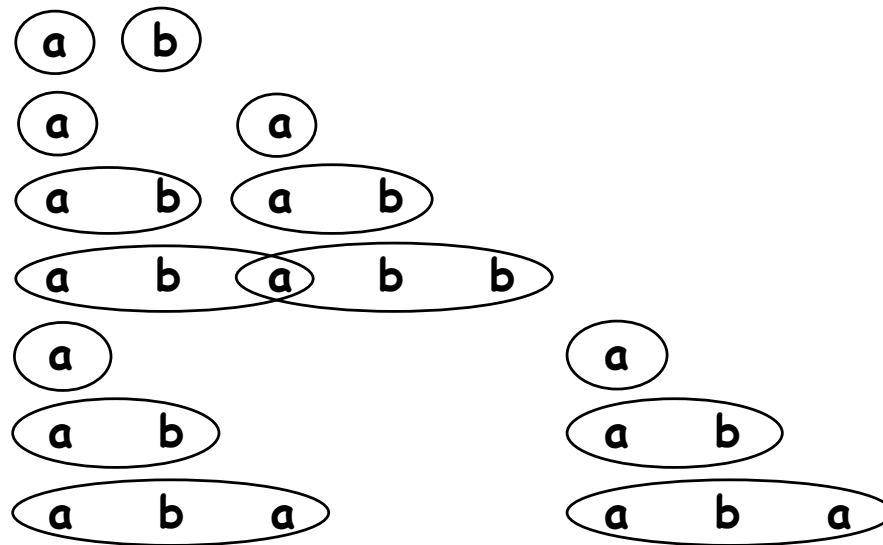


j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0	1	2	0	1	2				




Exemplo: cálculo da função de falha

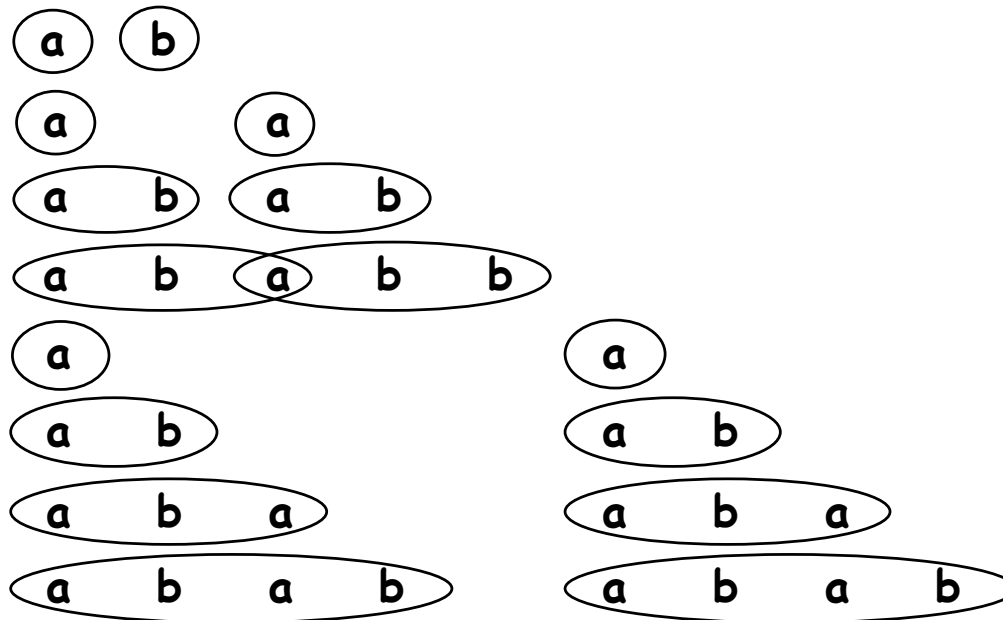
j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0	1	2	0	1	2	3			



Exemplo: cálculo da função de falha

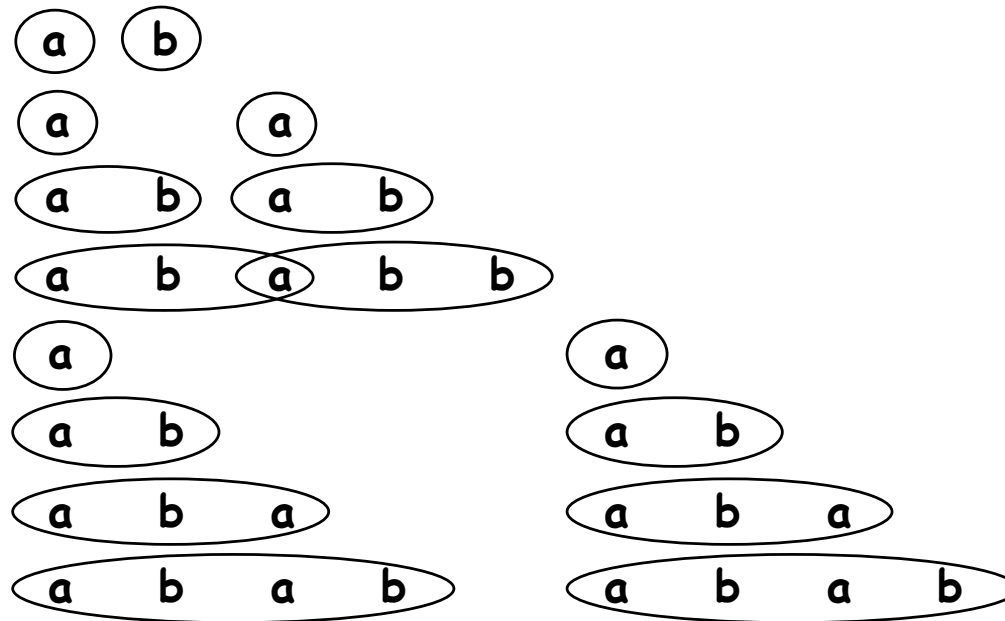


j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0	1	2	0	1	2	3			



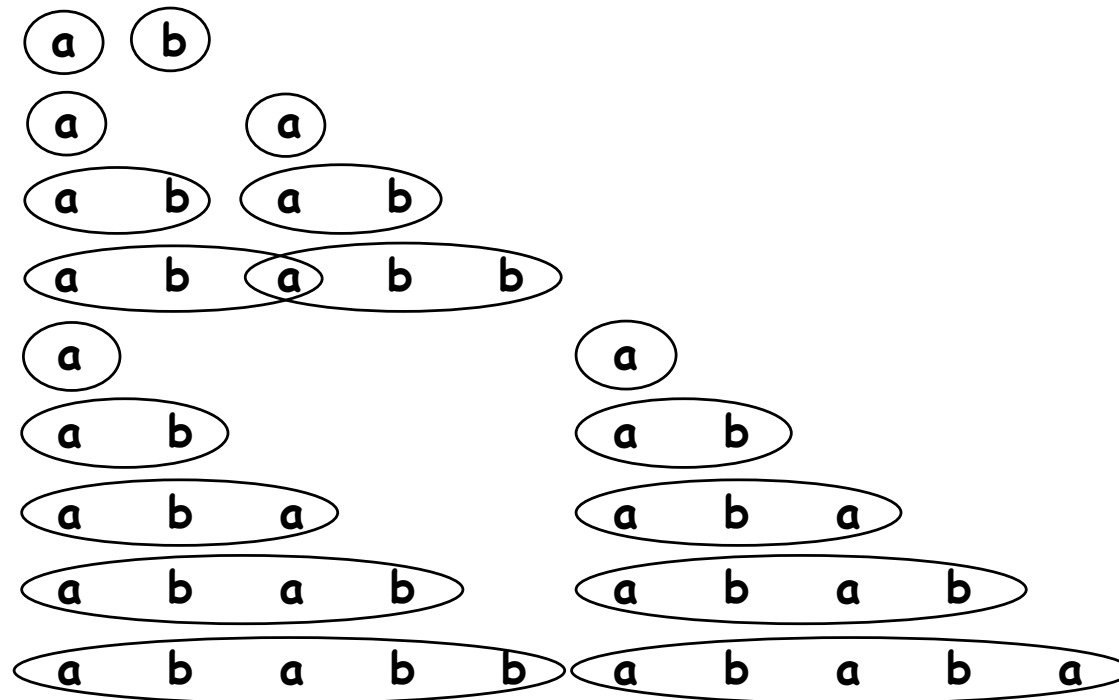
Exemplo: cálculo da função de falha

j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0	1	2	0	1	2	3	4		



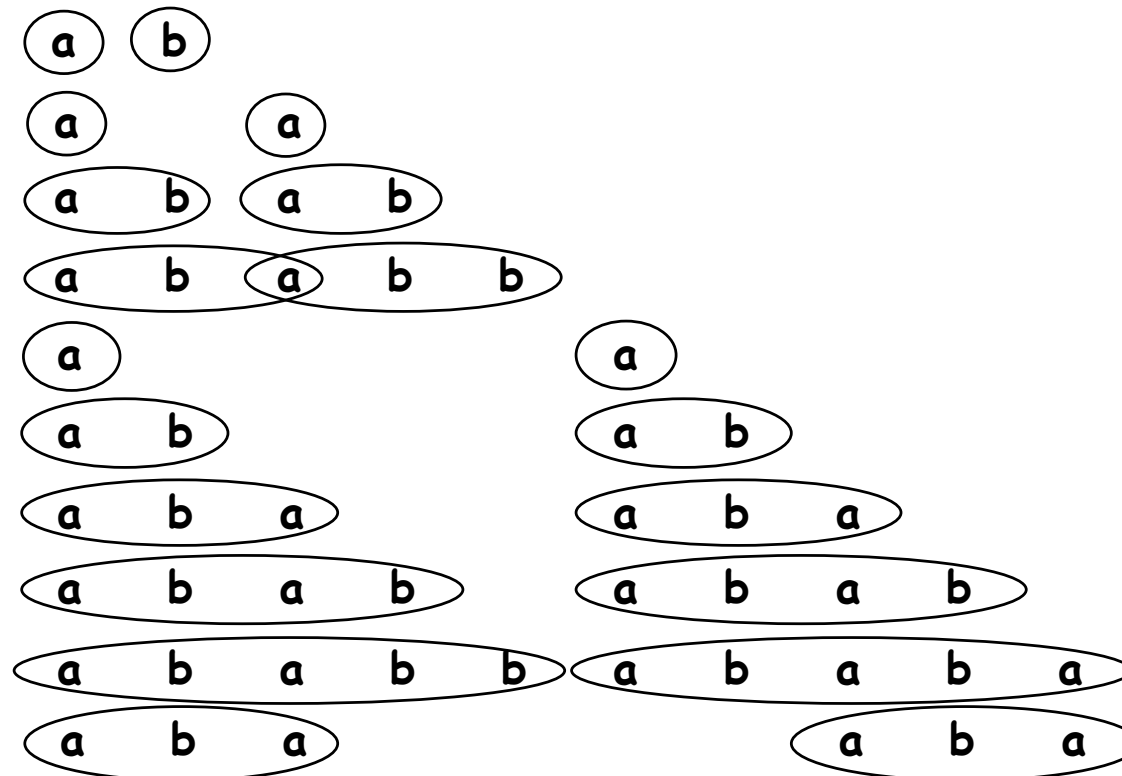
Exemplo: cálculo da função de falha

j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0	1	2	0	1	2	3	4		



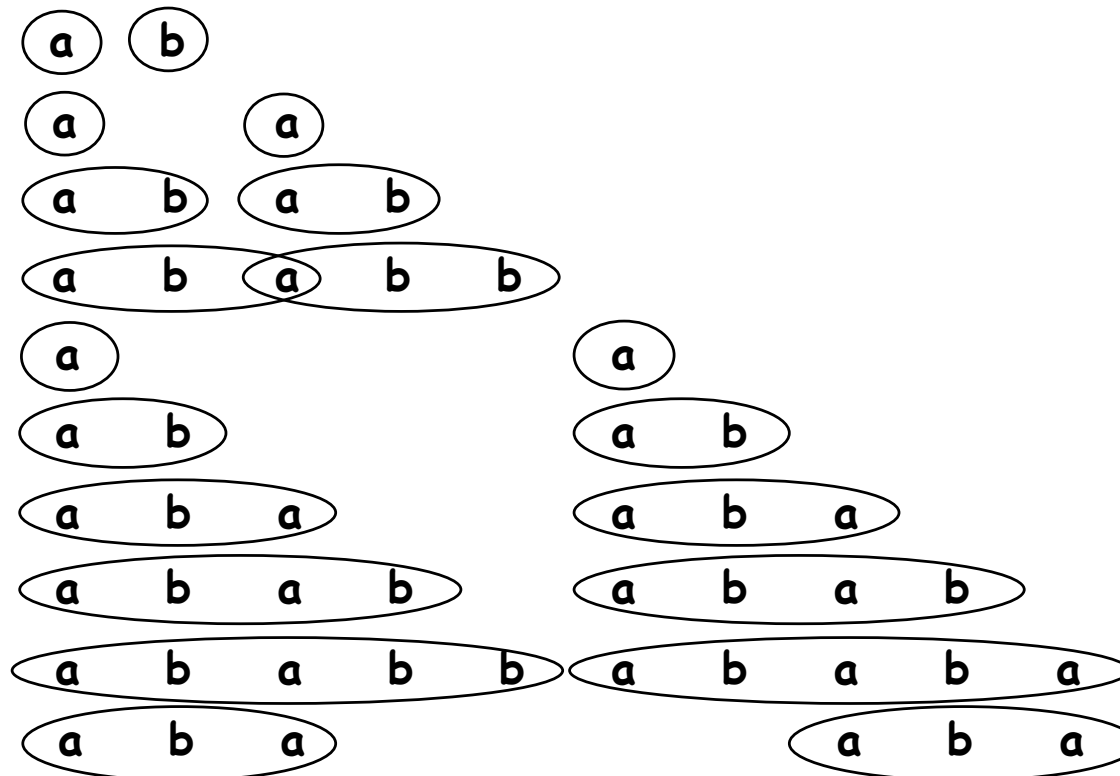
Exemplo: cálculo da função de falha

j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0	1	2	0	1	2	3	4		




Exemplo: cálculo da função de falha

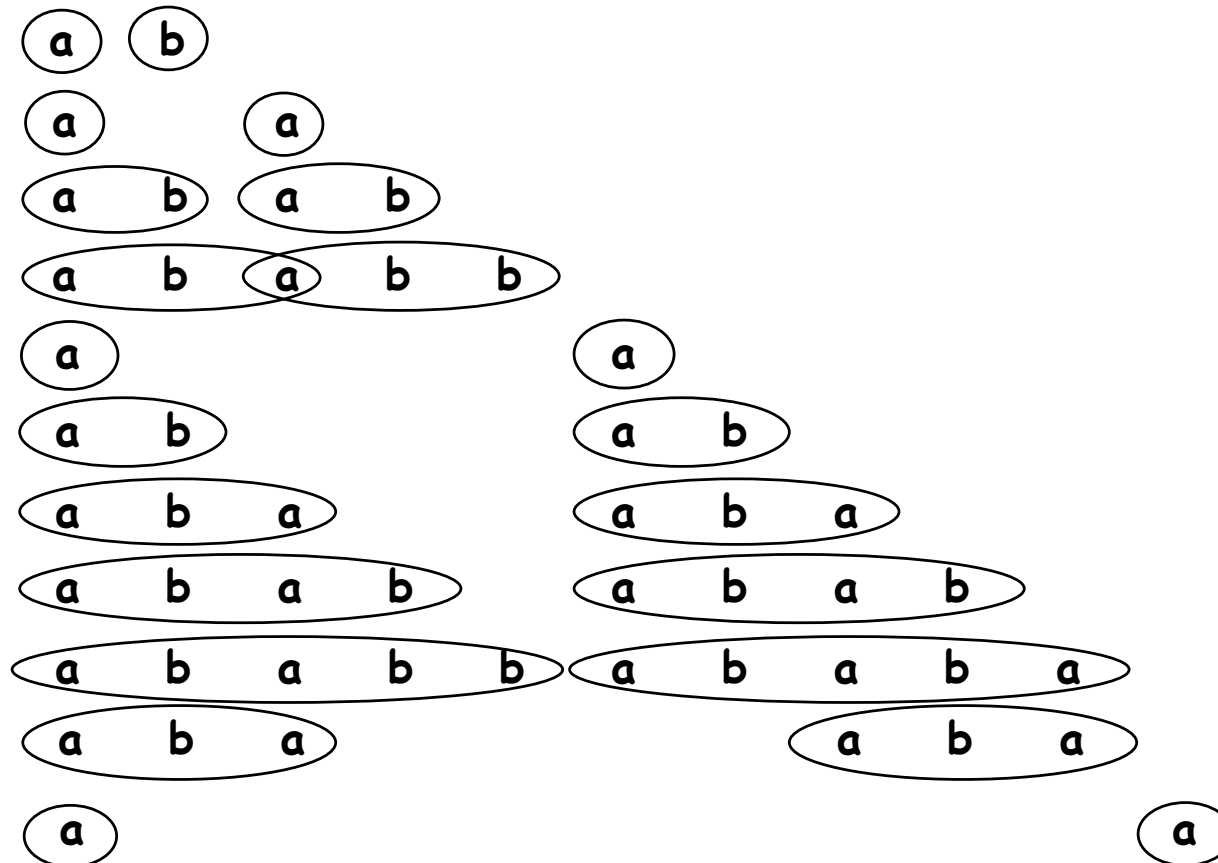
j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0	1	2	0	1	2	3	4	3	



Exemplo: cálculo da função de falha

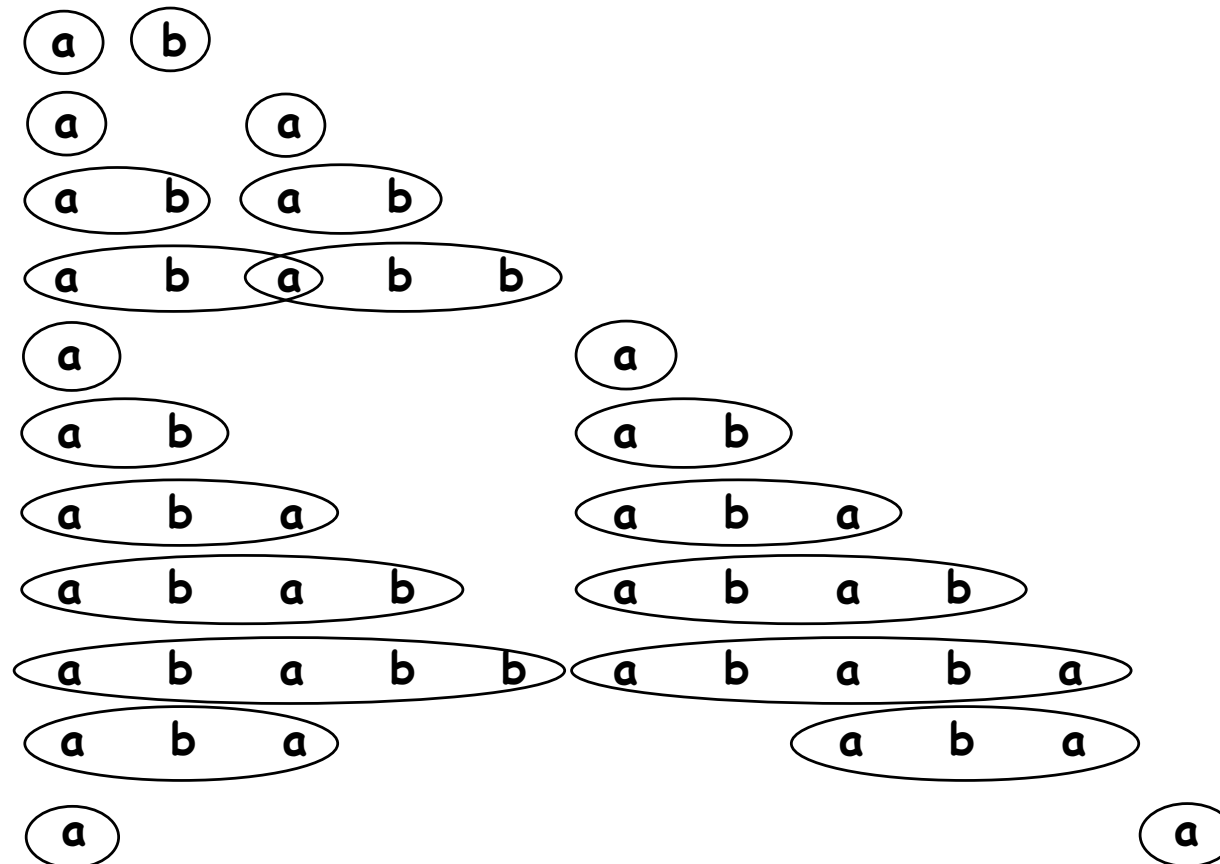


j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0	1	2	0	1	2	3	4	3	

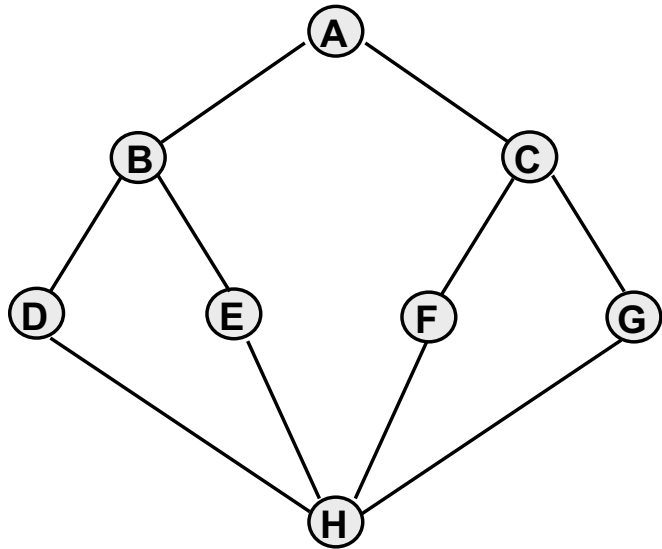


Exemplo: cálculo da função de falha

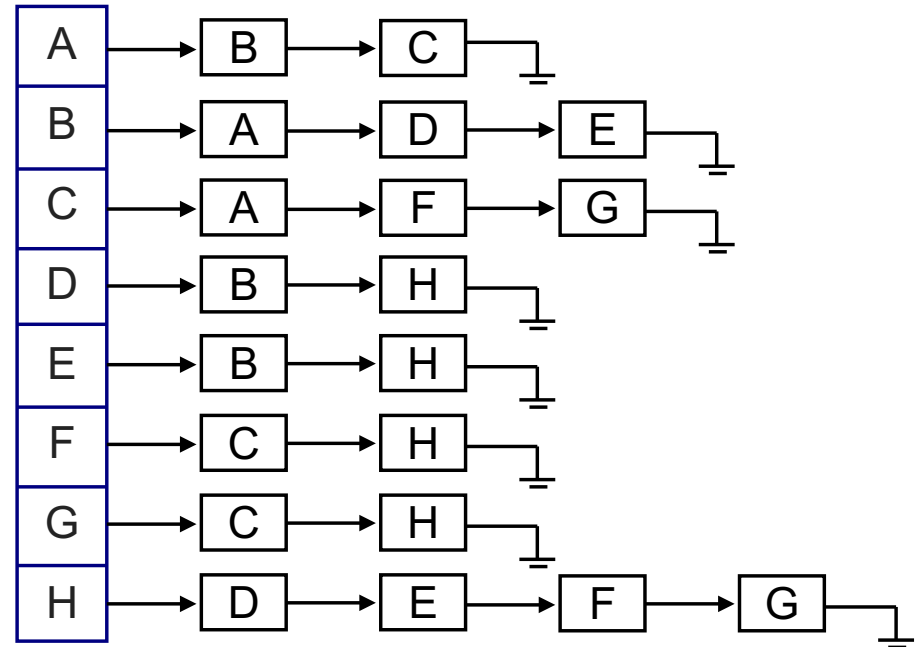
j	0	1	2	3	4	5	6	7	8	9	10
$P[j]$	a	b	a	b	b	a	b	a	b	a	a
$F[j]$	0	0	1	2	0	1	2	3	4	3	1



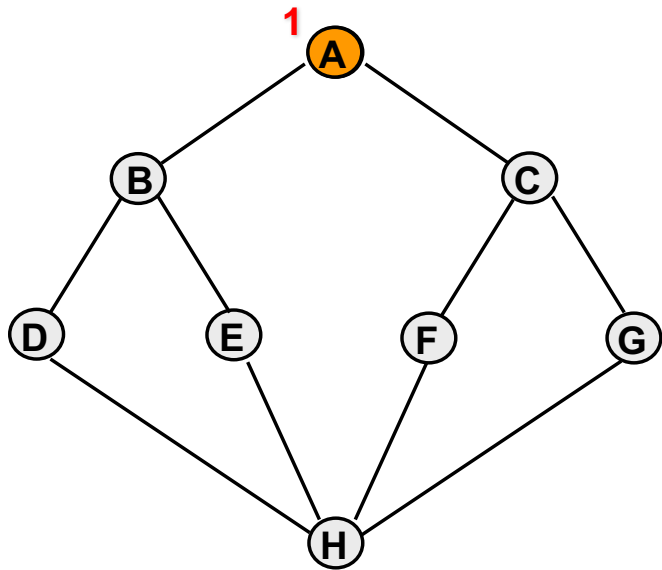
Exemplo



- não visitado
- visitado

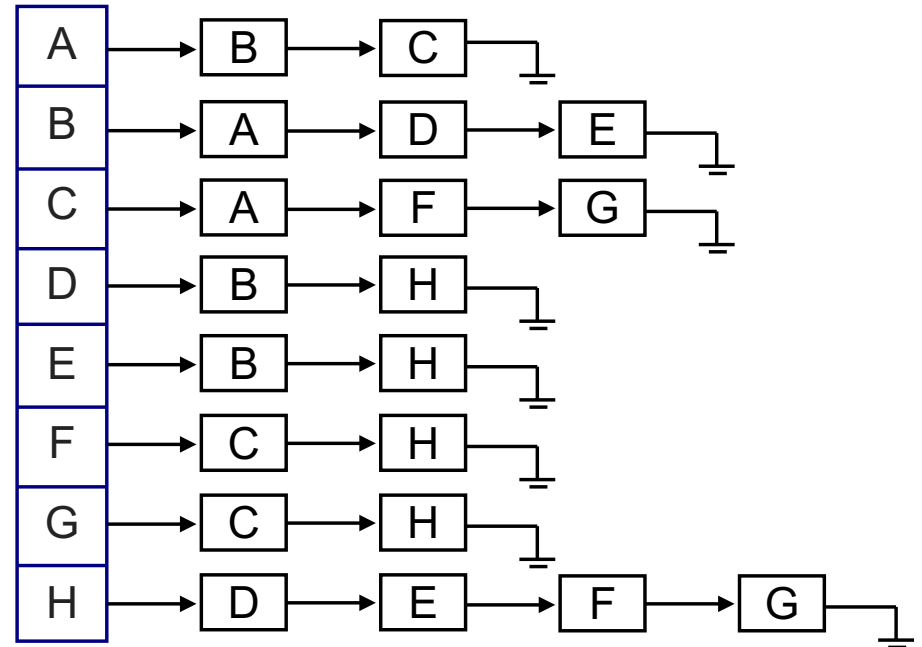


Exemplo

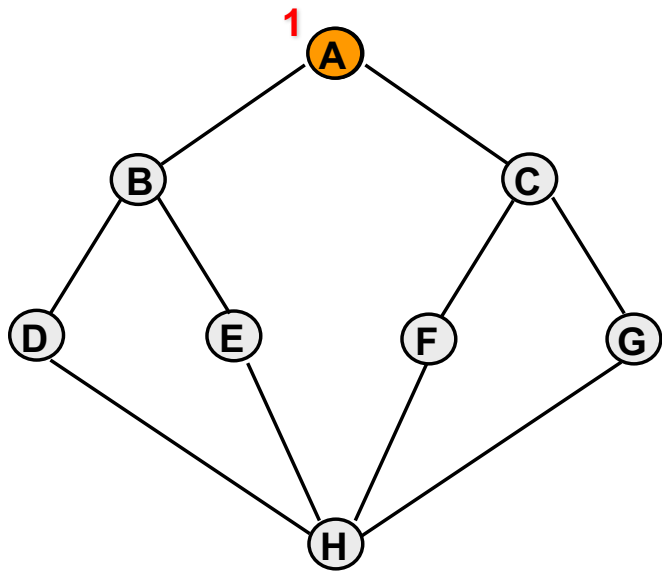


○ não visitado

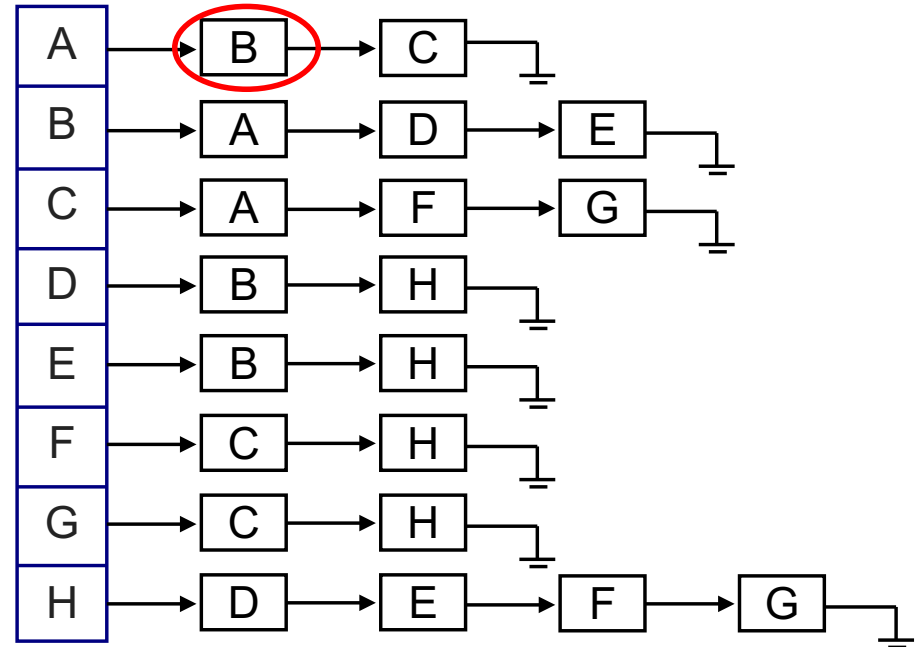
● visitado



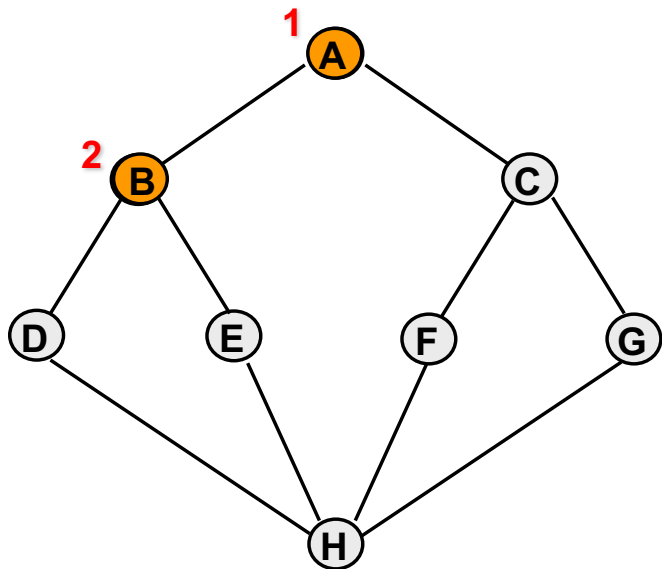
Exemplo



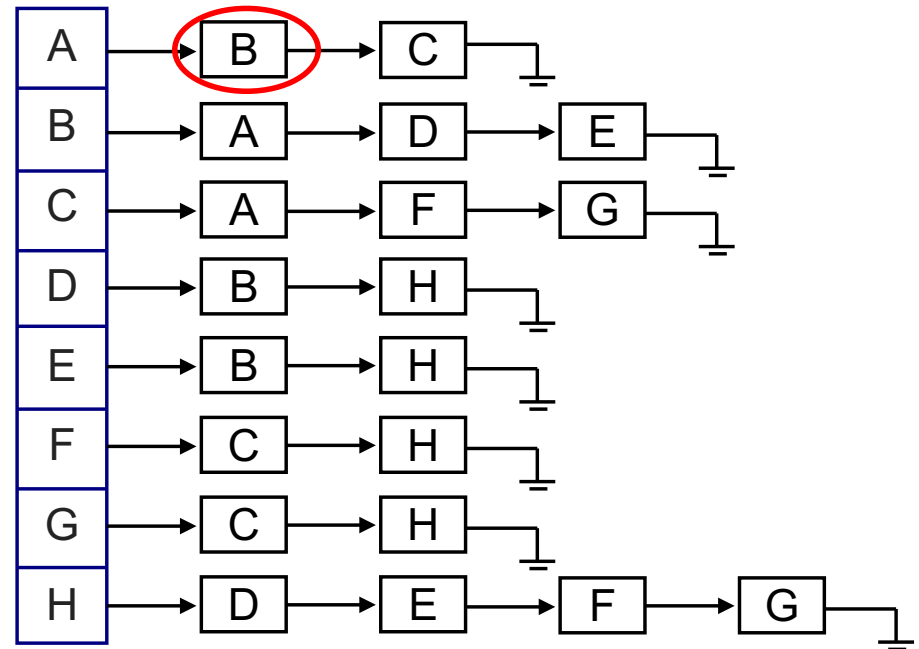
- não visitado
- visitado



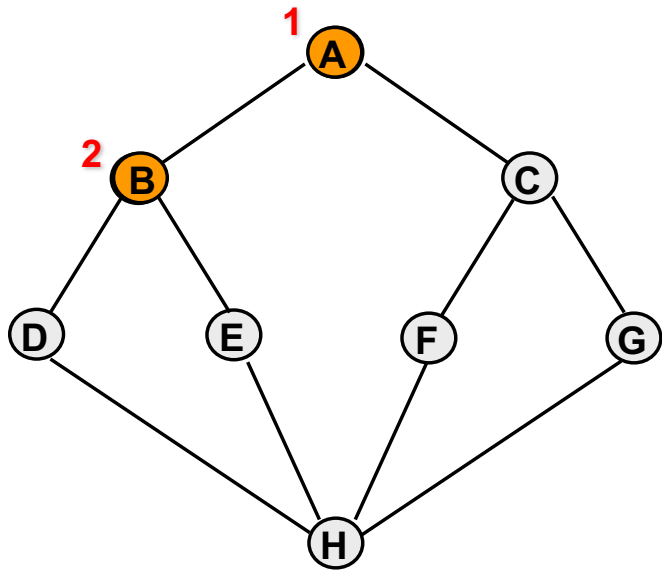
Exemplo



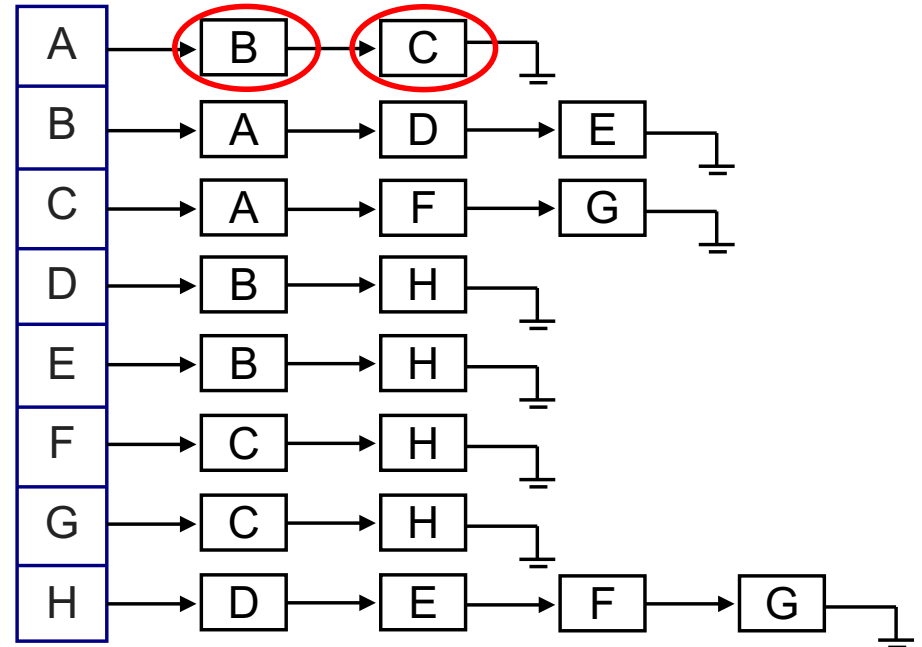
- não visitado
- visitado



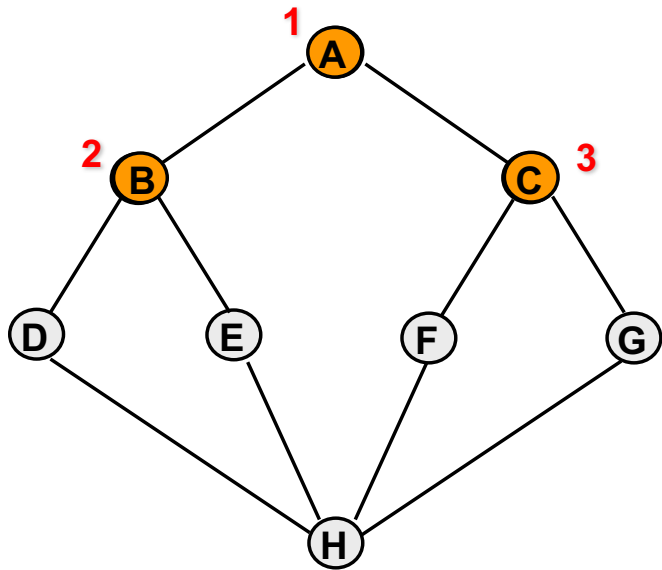
Exemplo



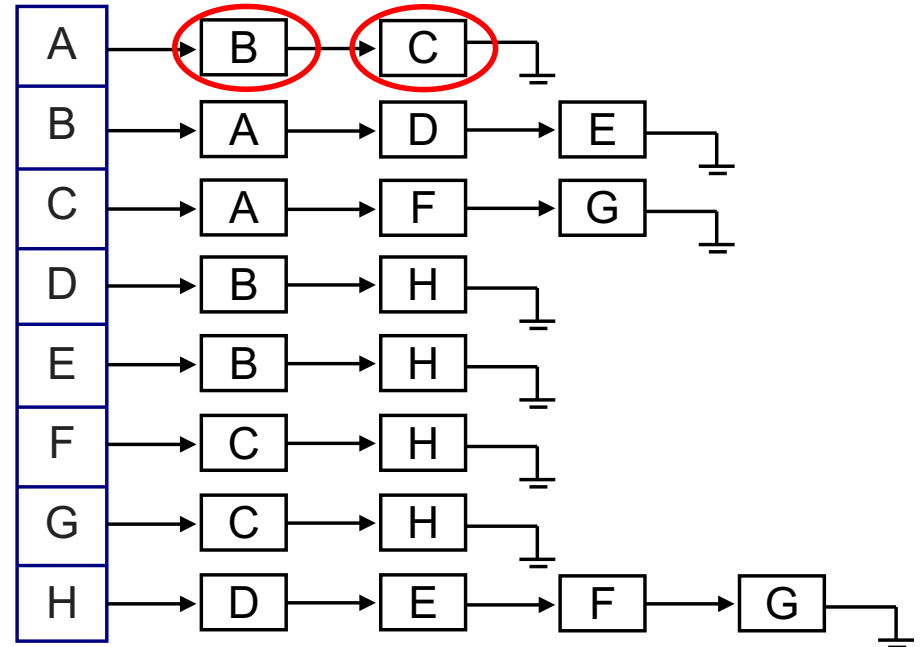
- não visitado
- visitado



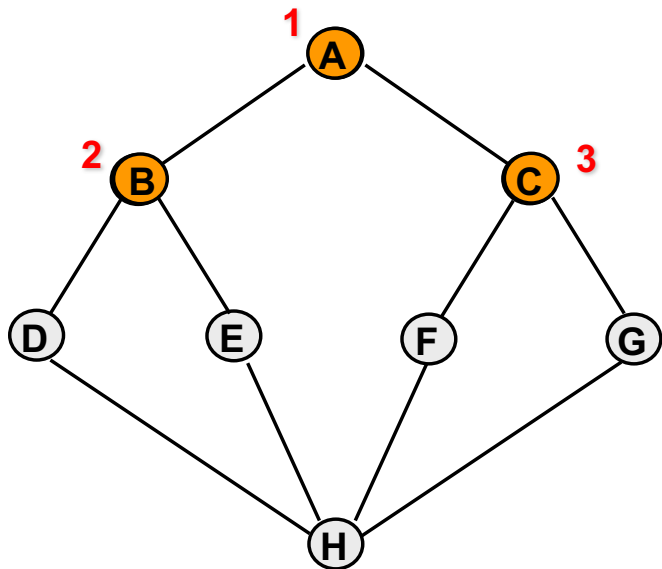
Exemplo



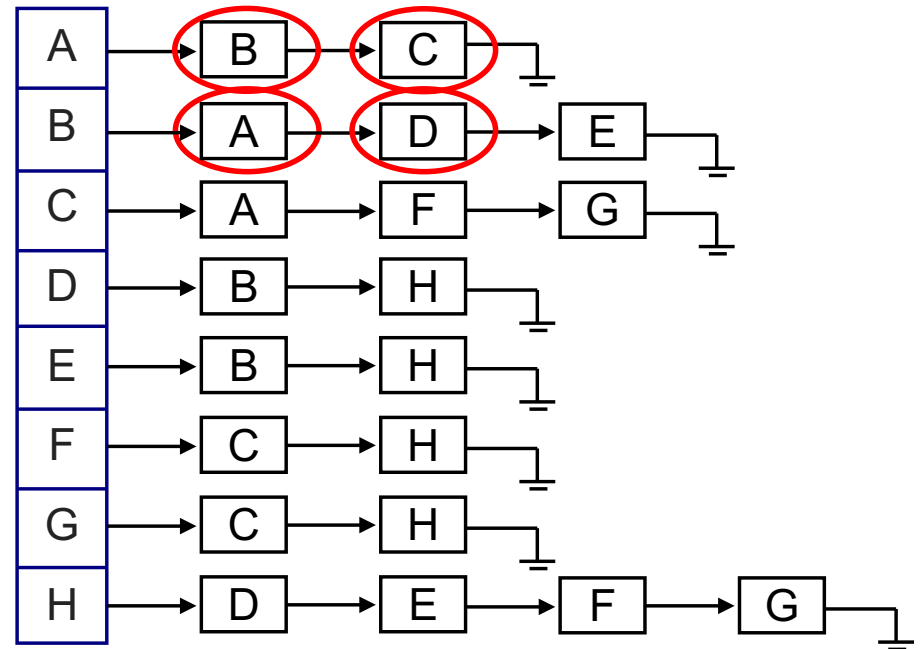
- não visitado
- visitado



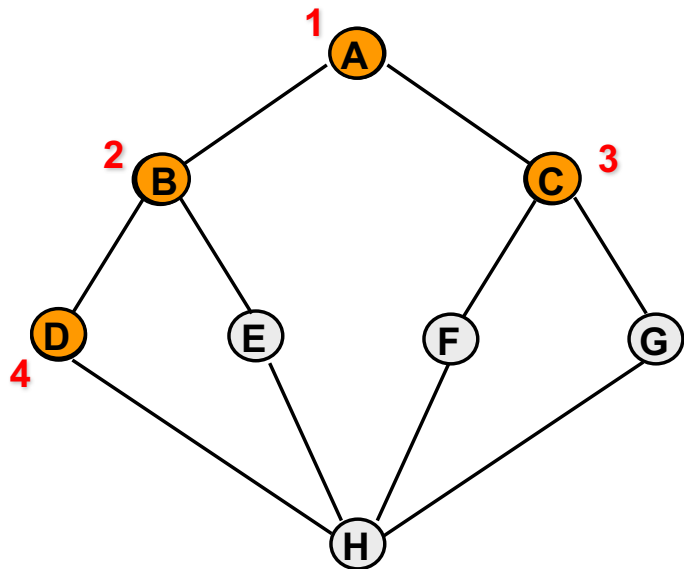
Exemplo



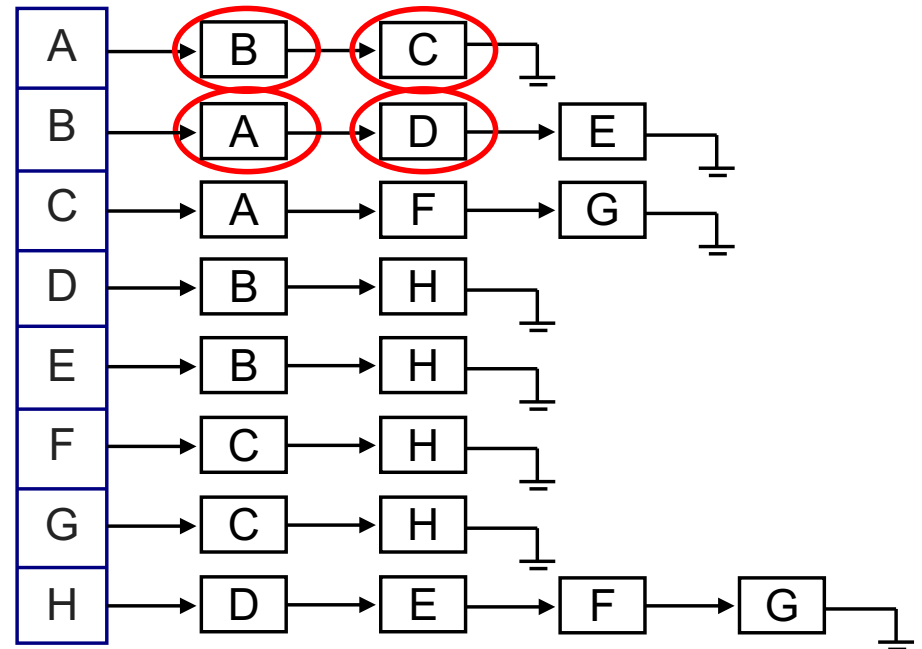
- não visitado
- visitado



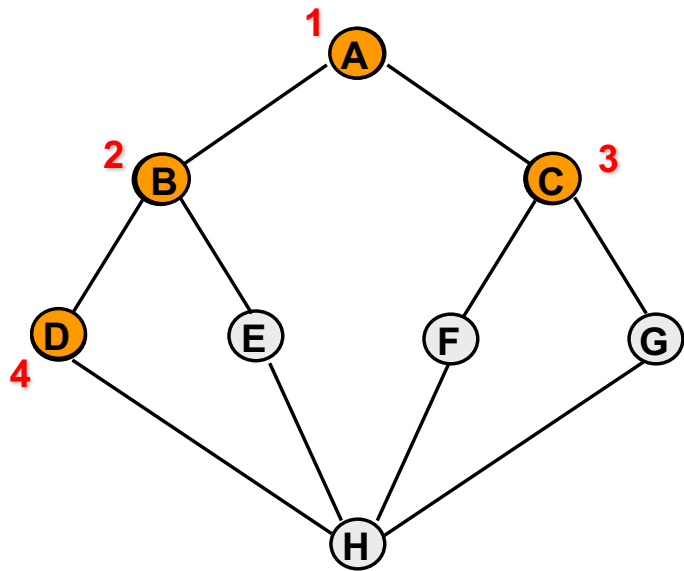
Exemplo



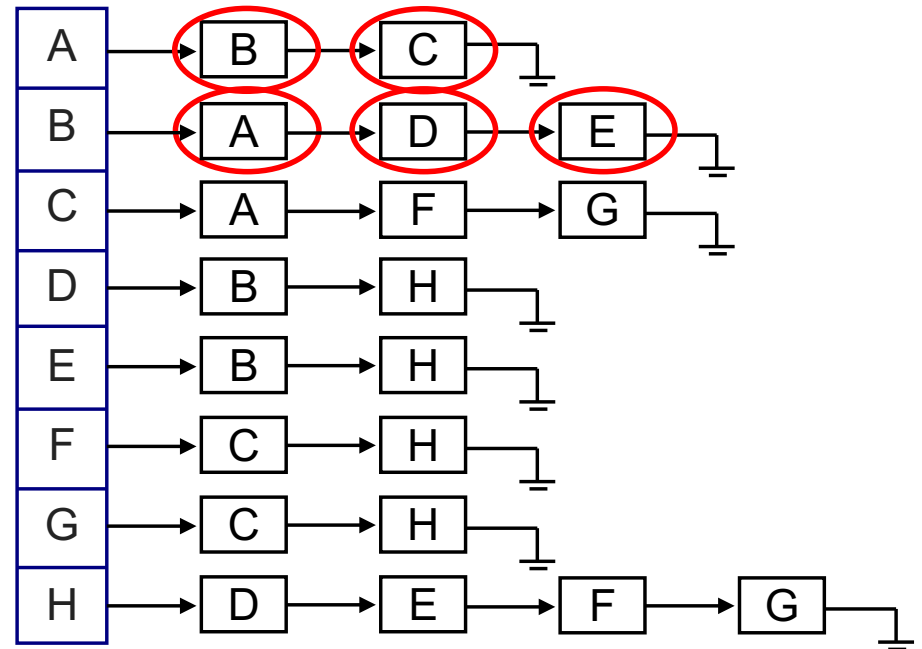
- não visitado
- visitado



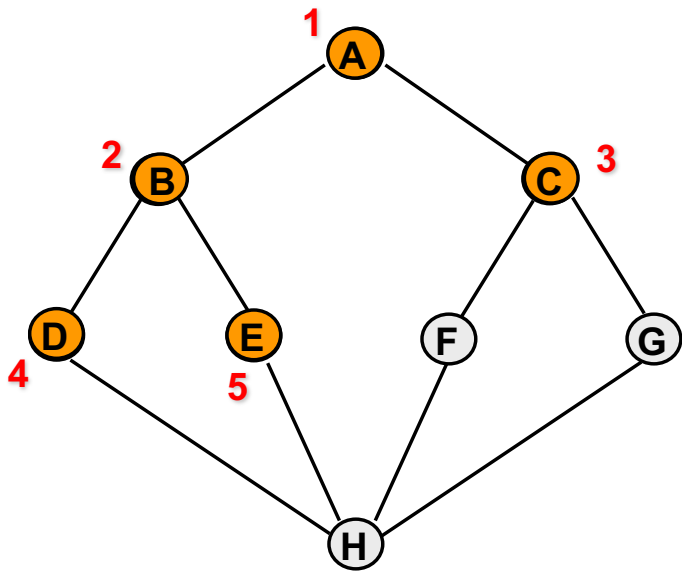
Exemplo



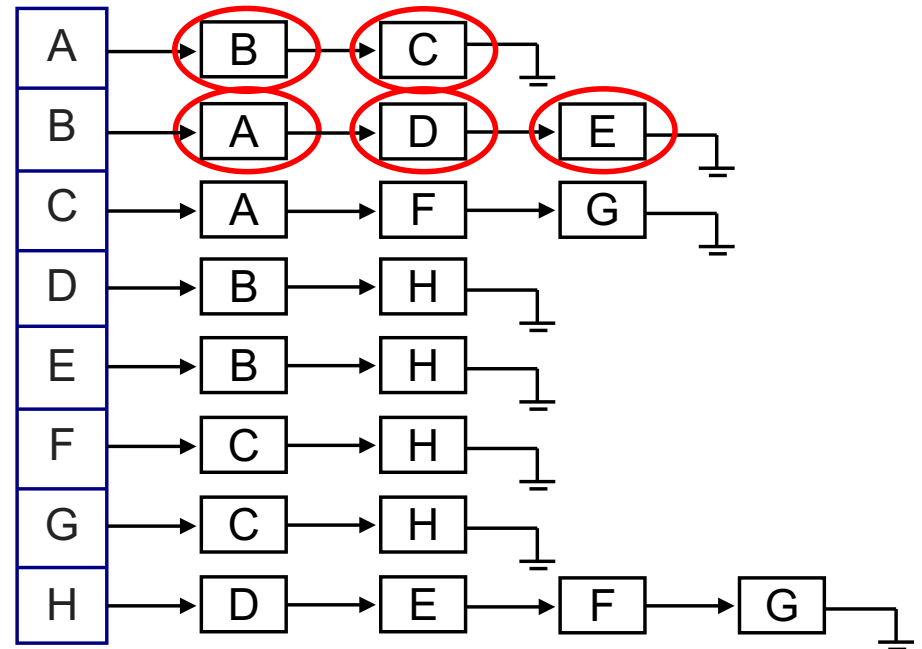
- não visitado
- visitado



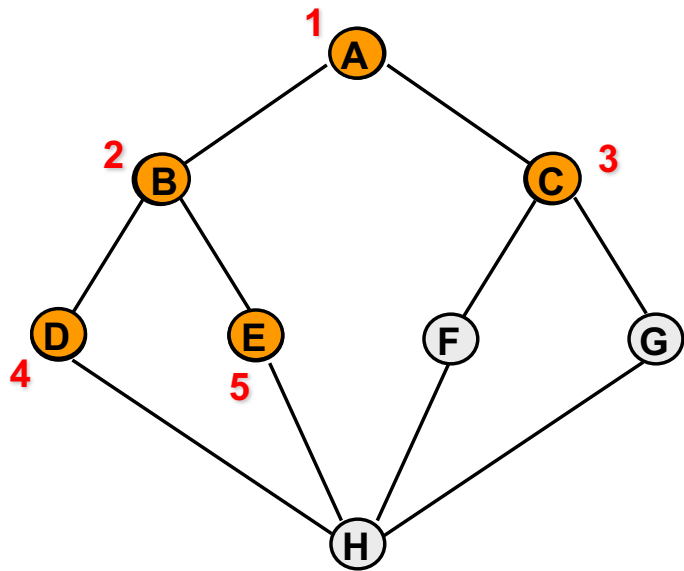
Exemplo



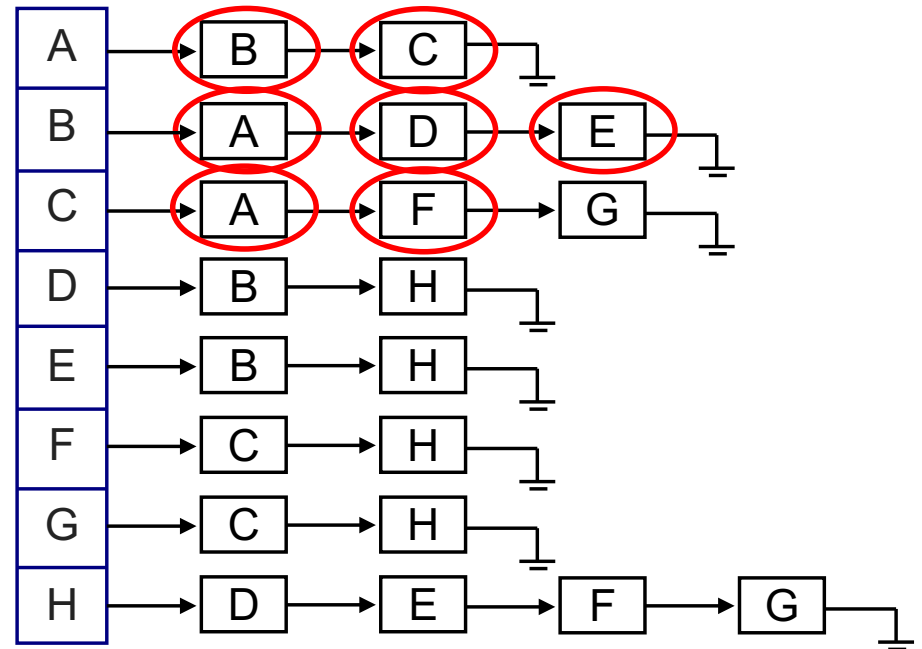
- não visitado
- visitado



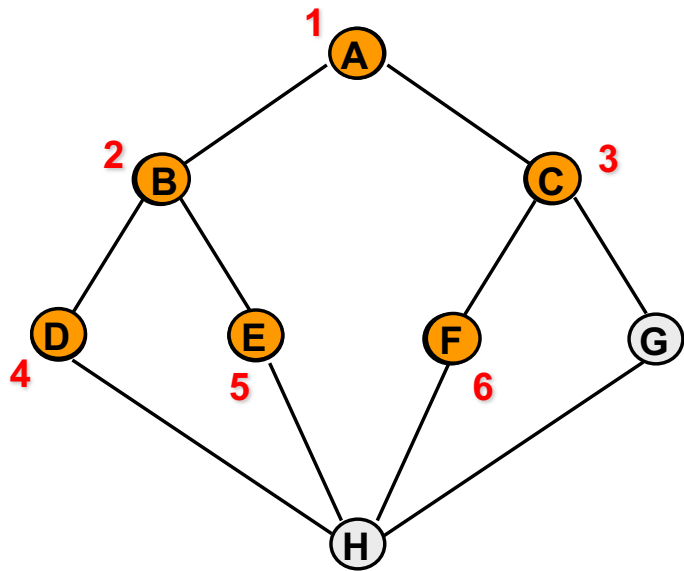
Exemplo



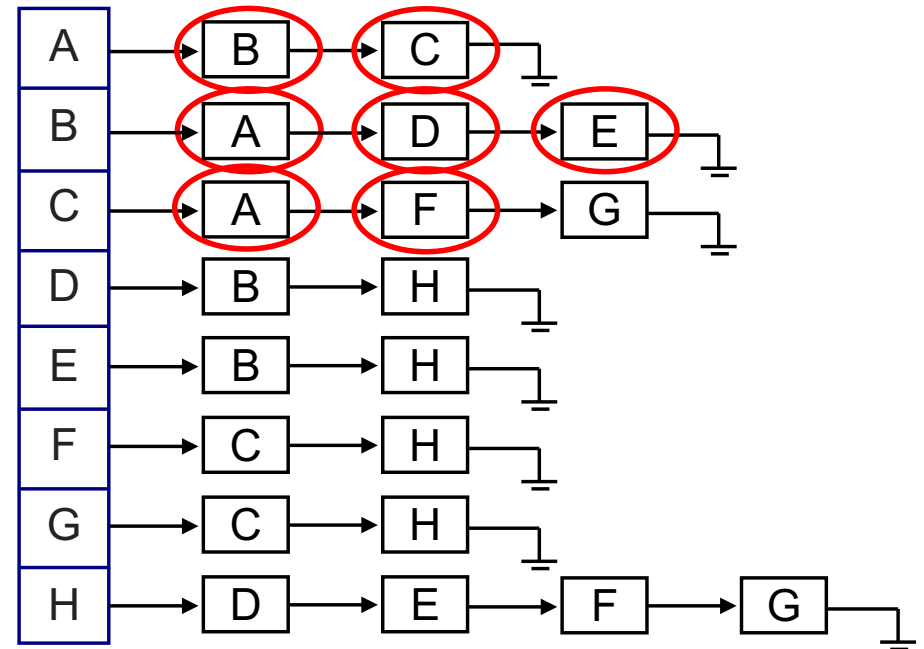
- não visitado
- visitado



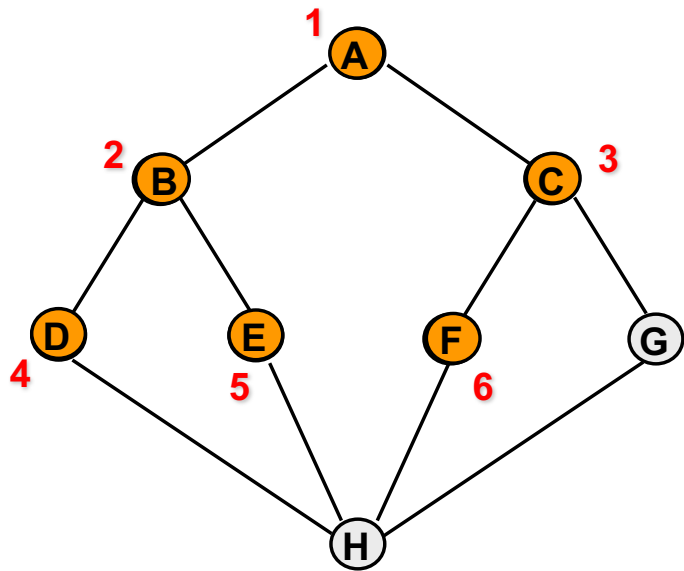
Exemplo



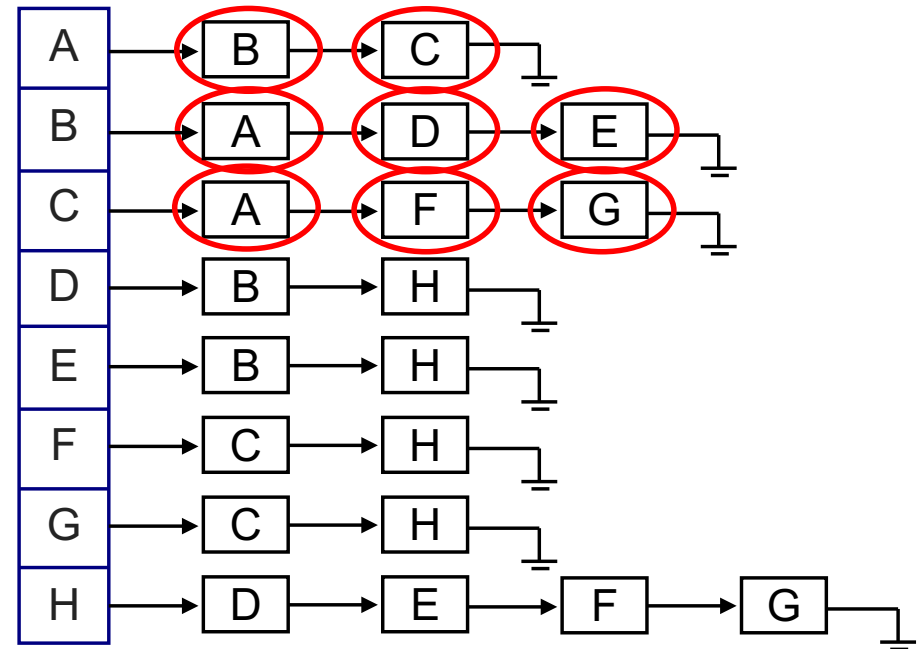
- não visitado
- visitado



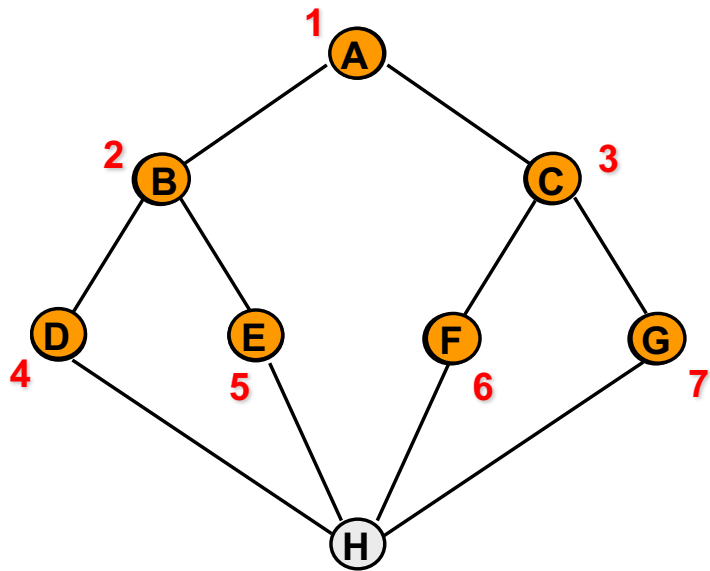
Exemplo



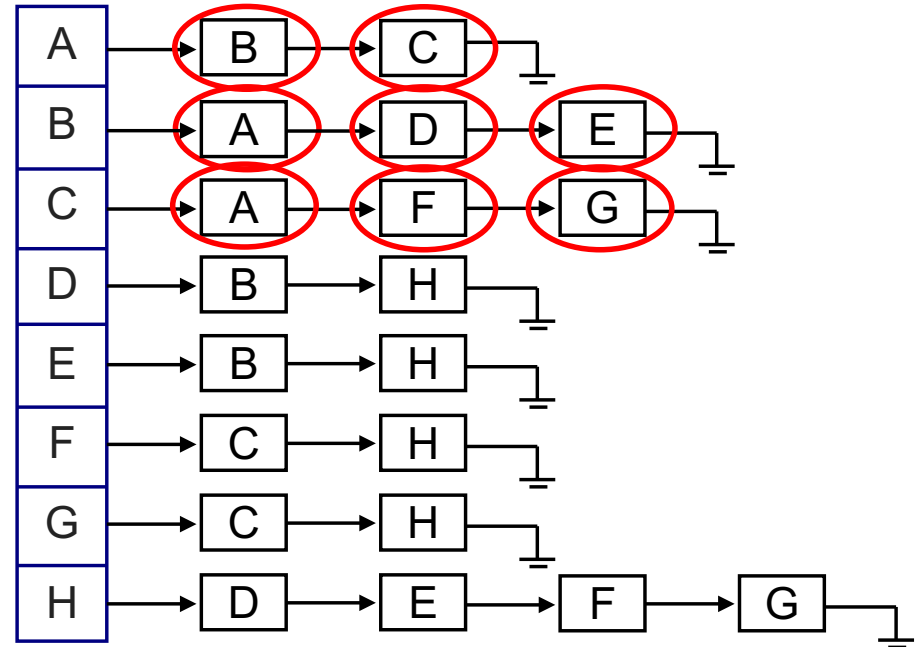
- não visitado
- visitado



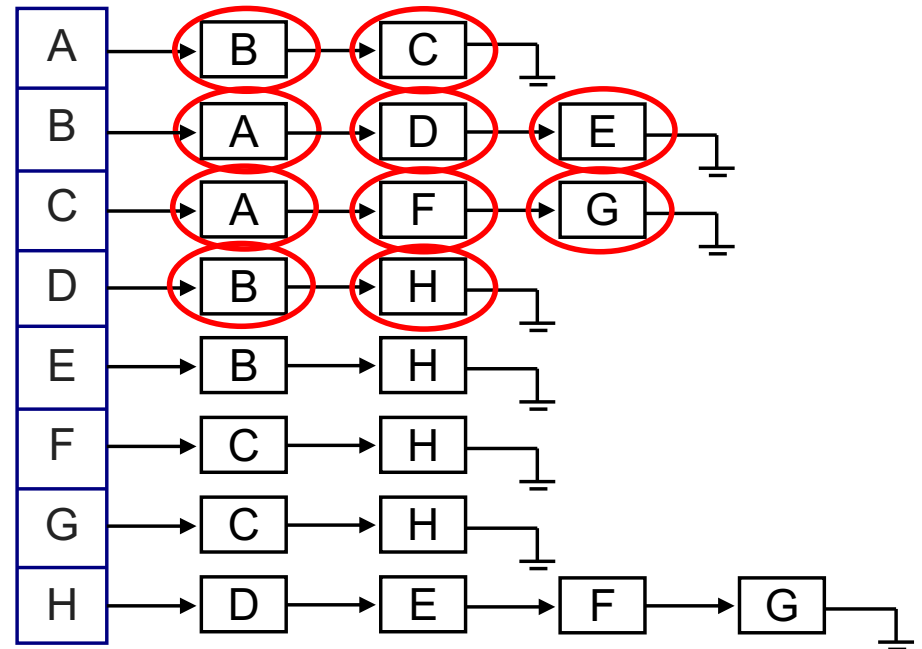
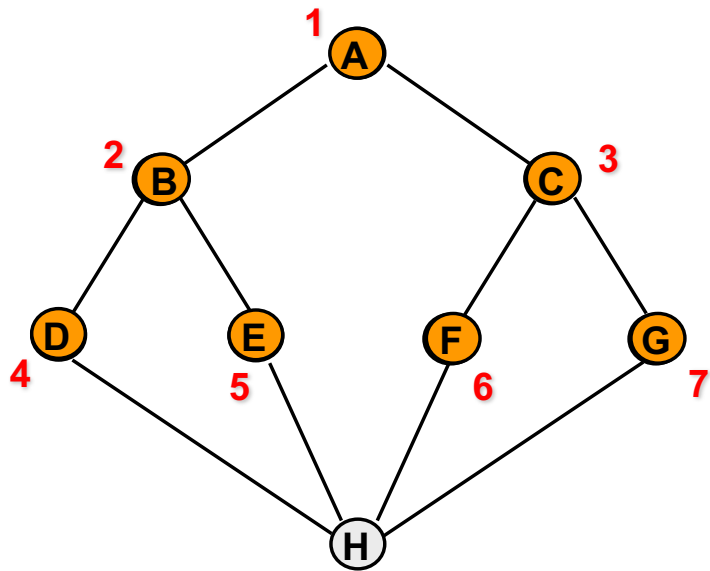
Exemplo



- não visitado
- visitado

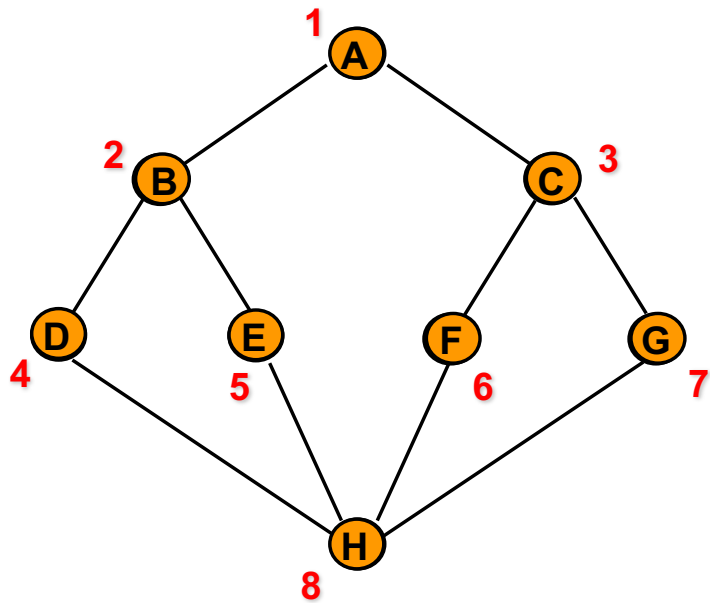


Exemplo

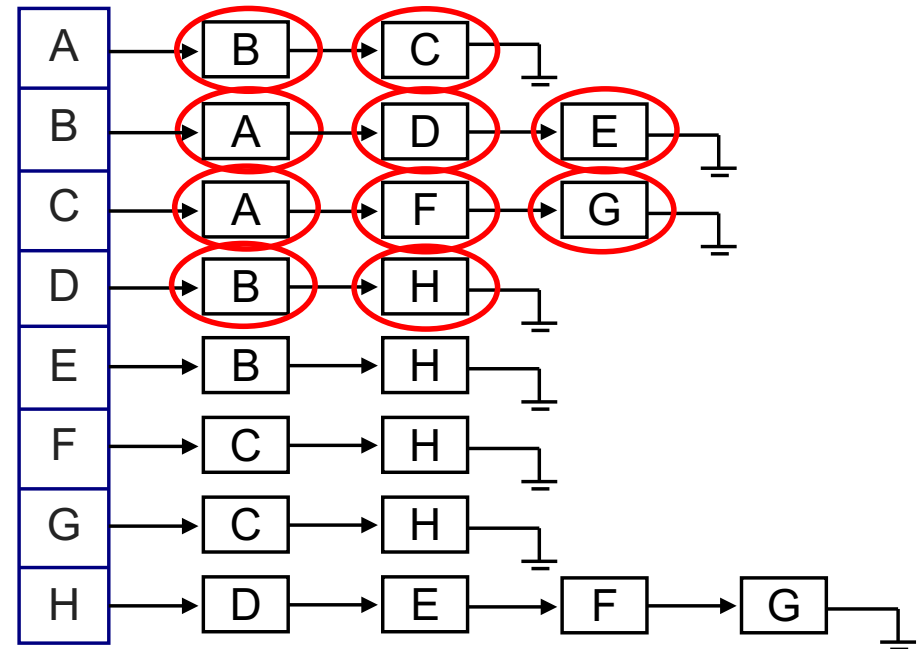


- não visitado
- visitado

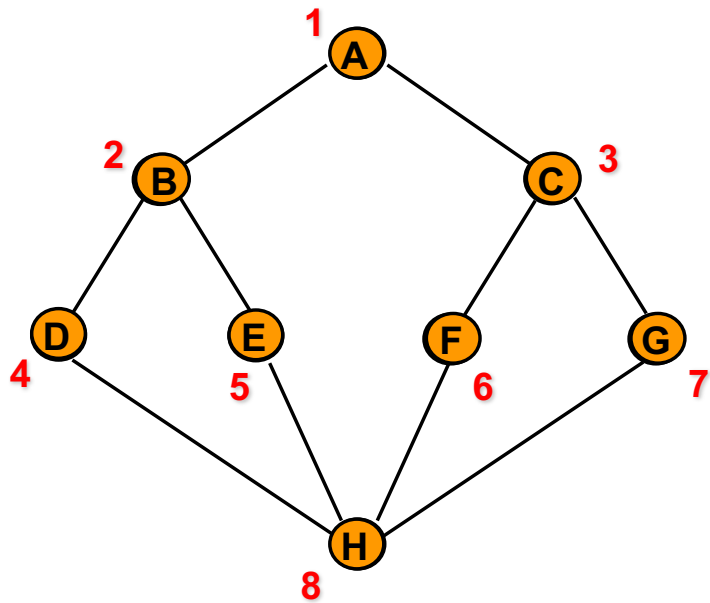
Exemplo



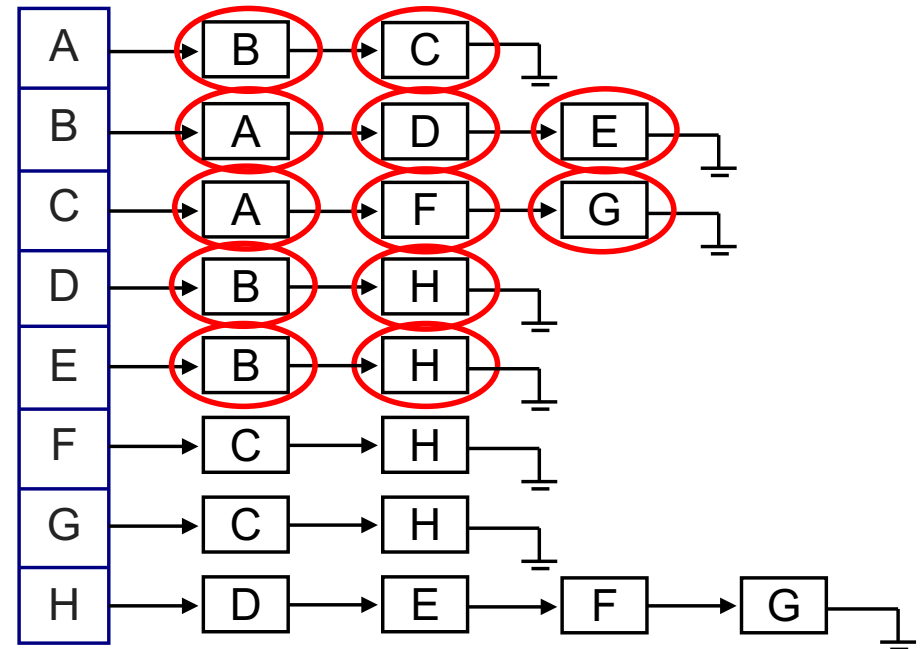
- não visitado
- visitado



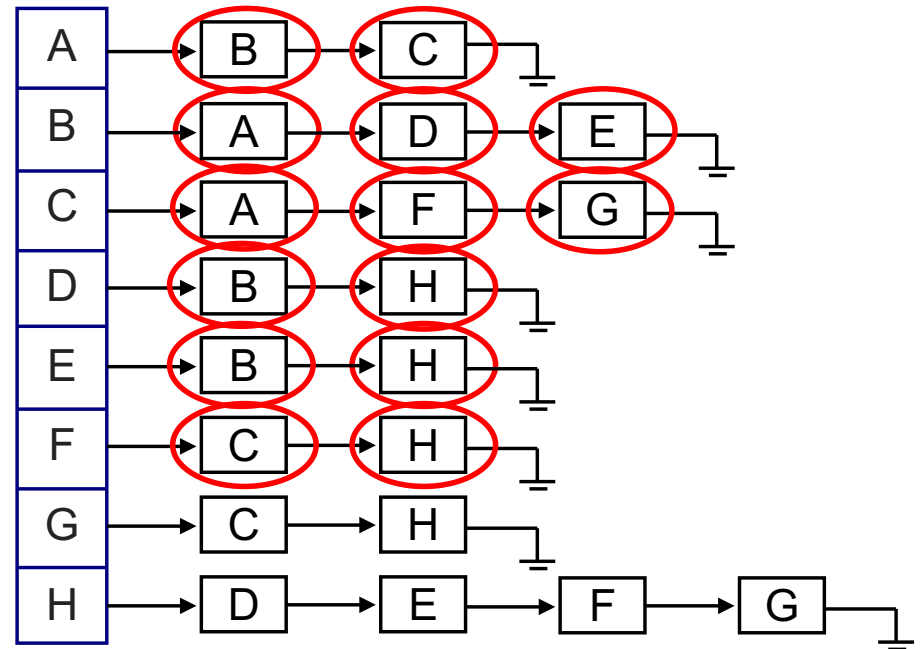
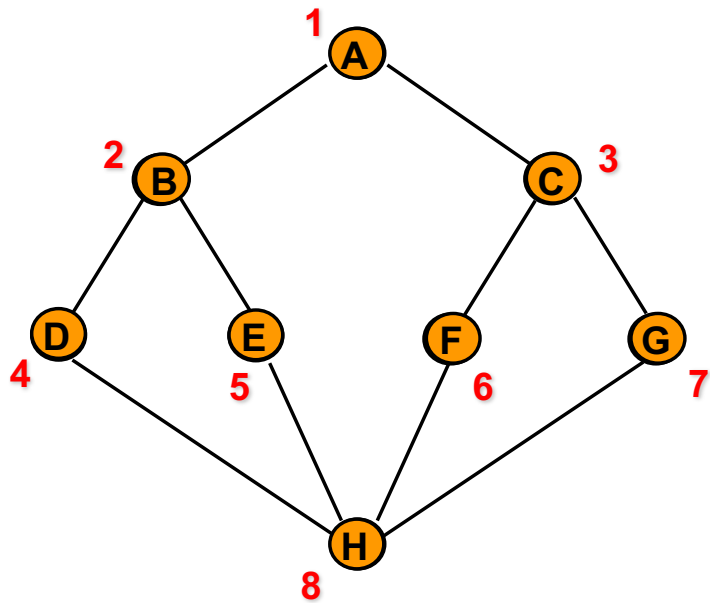
Exemplo



- não visitado
- visitado

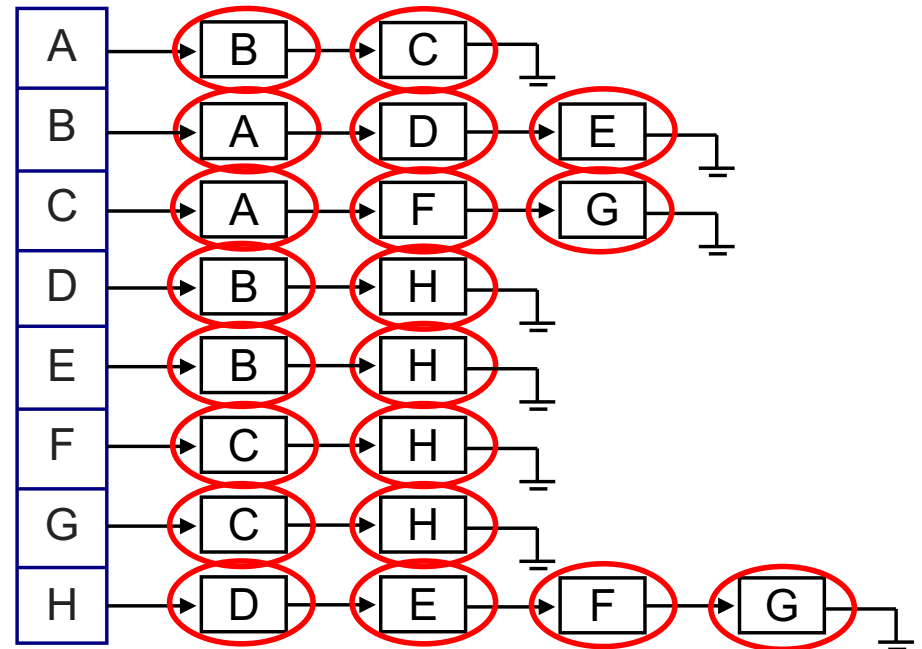
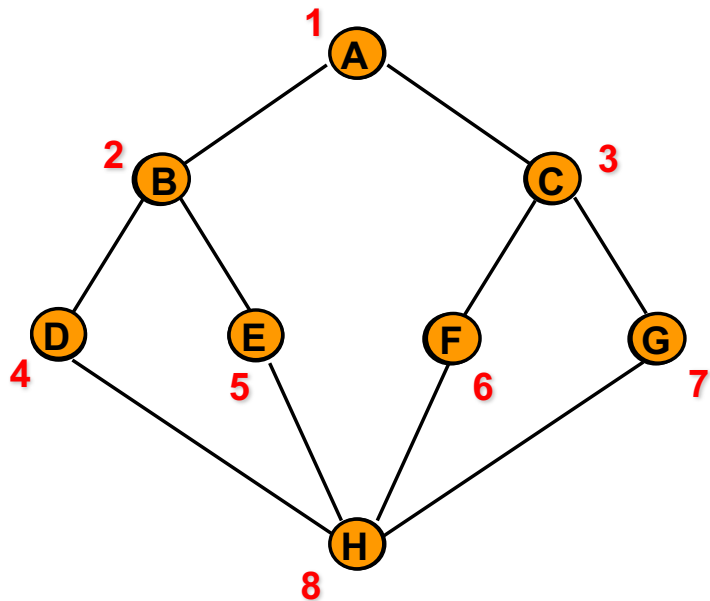


Exemplo

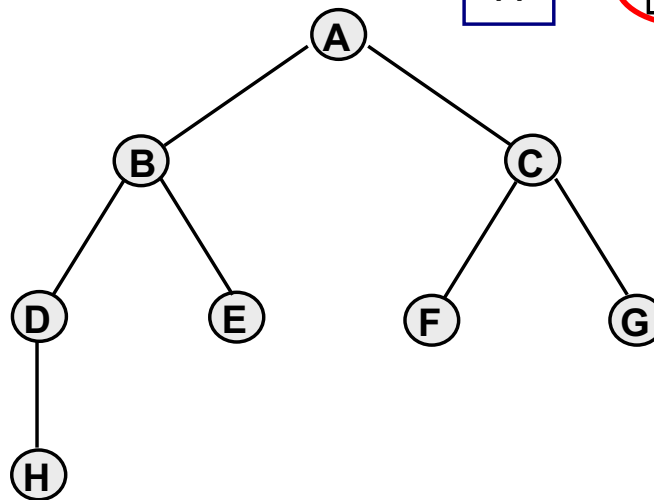


- não visitado
- visitado

Exemplo

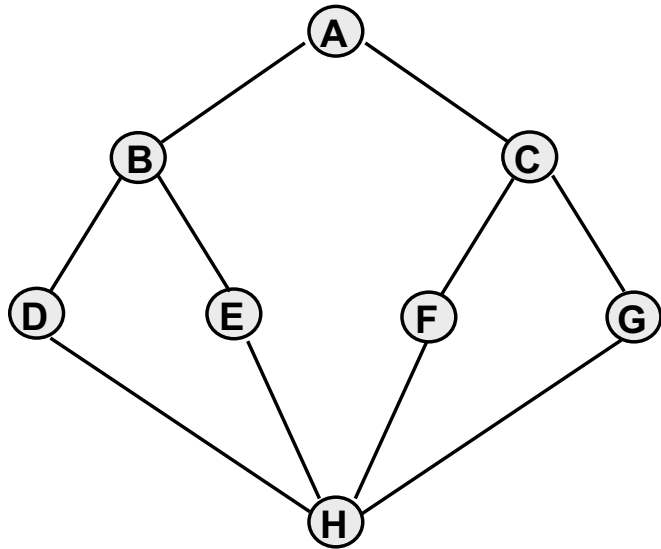


- não visitado
- visitado



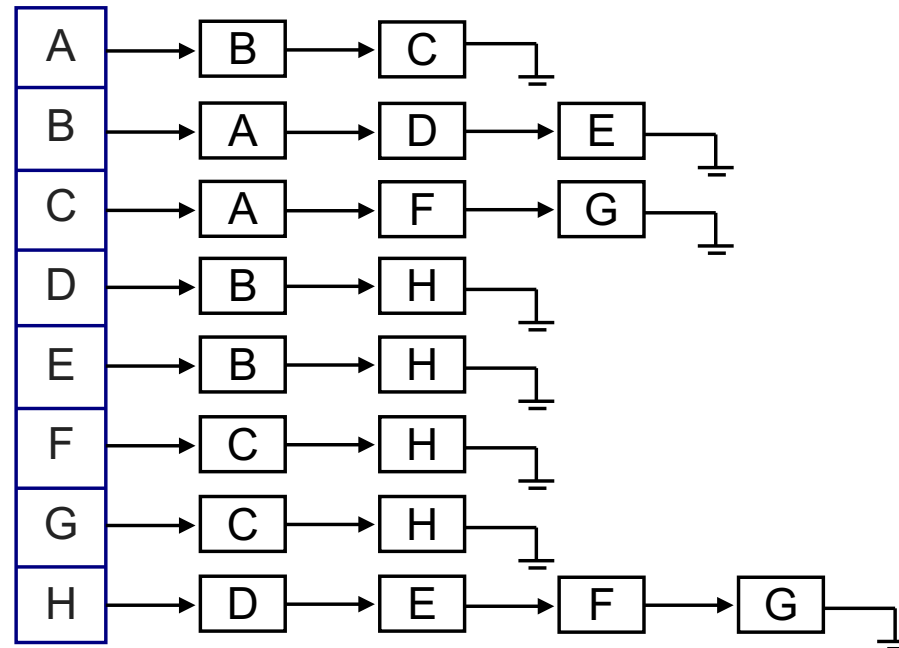
Árvore de exploração em largura

Exemplo

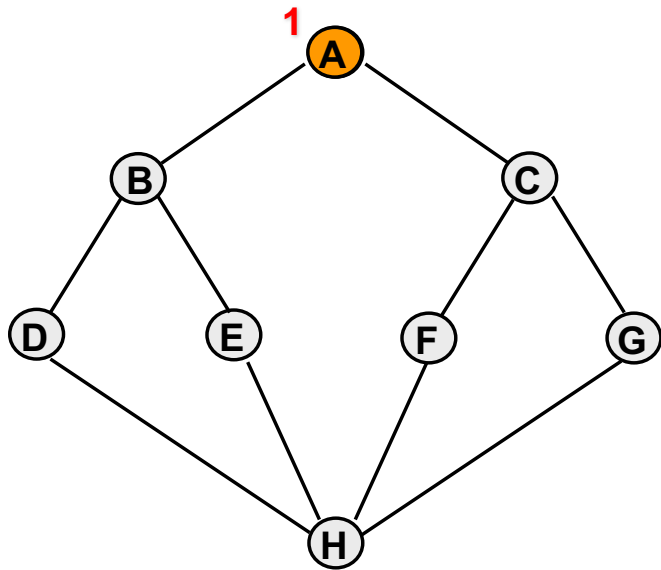


○ não visitado

● visitado

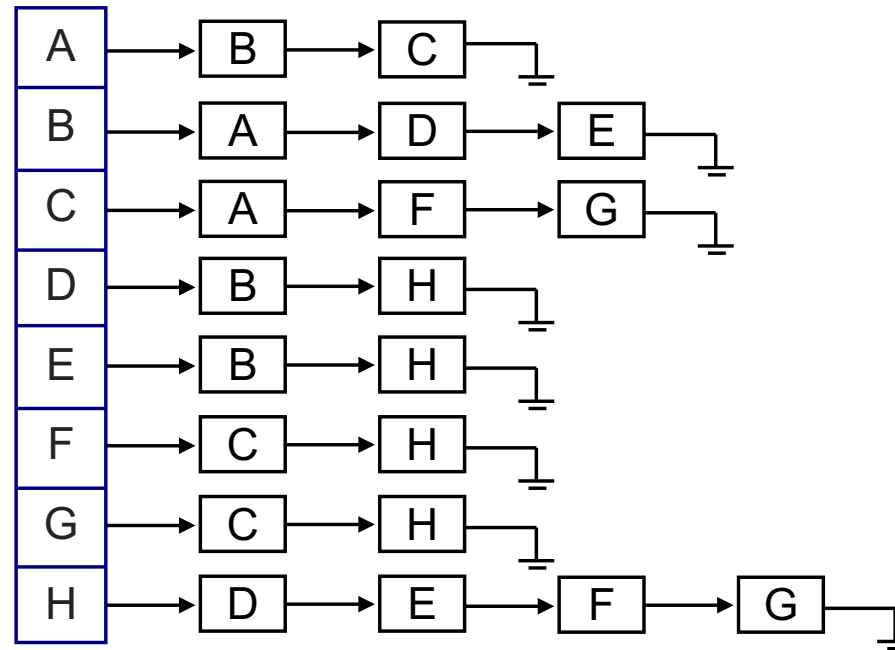


Exemplo

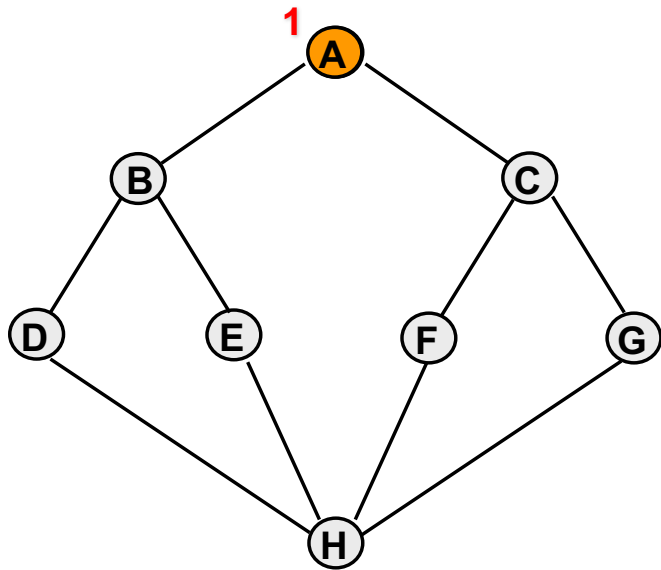


○ não visitado

● visitado

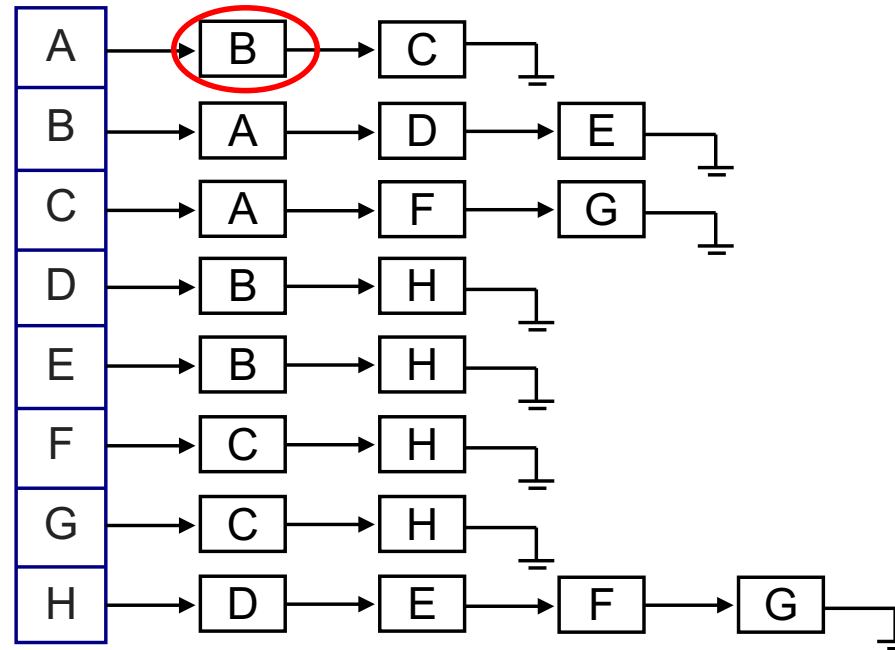


Exemplo

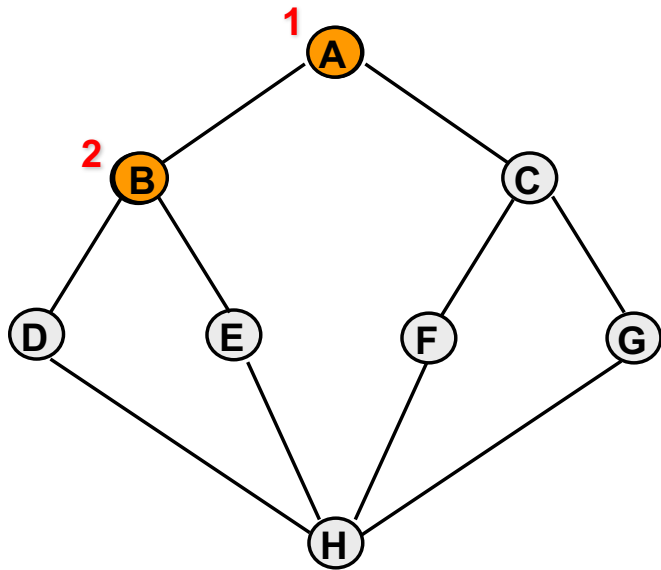


○ não visitado

● visitado

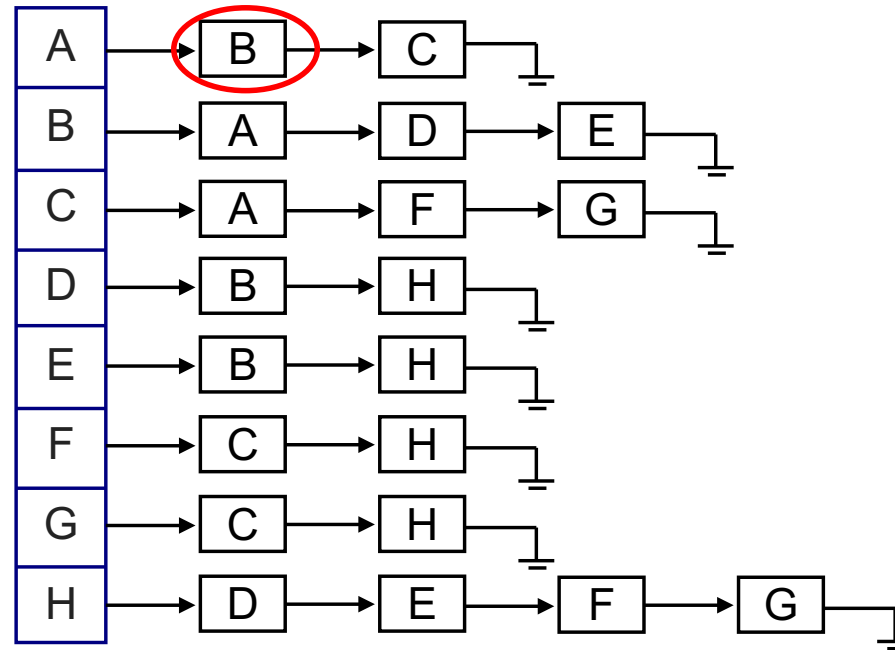


Exemplo

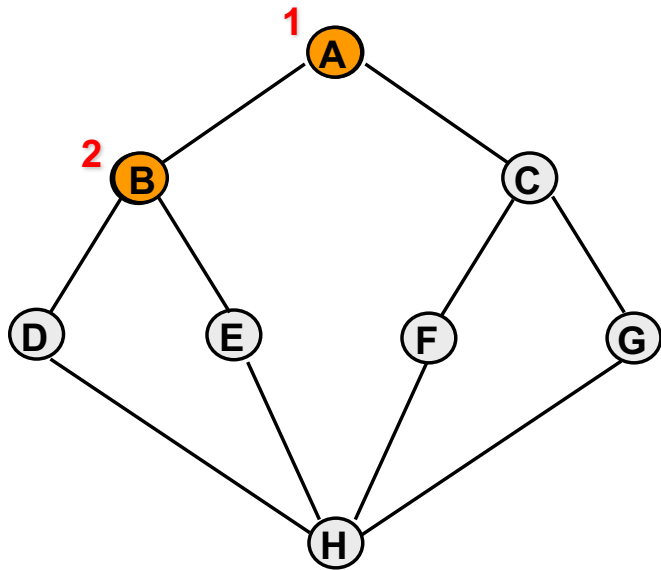


○ não visitado

● visitado

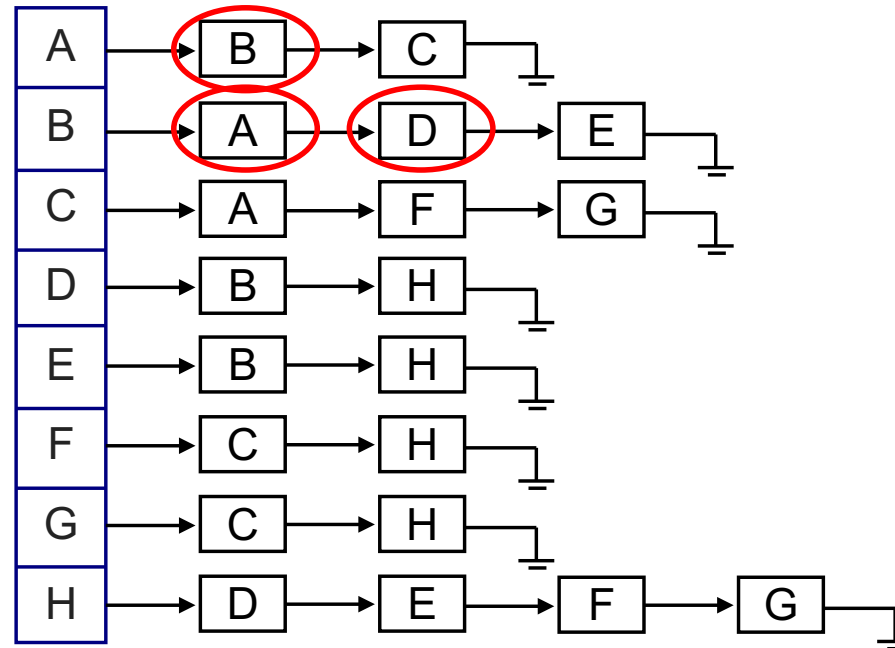


Exemplo

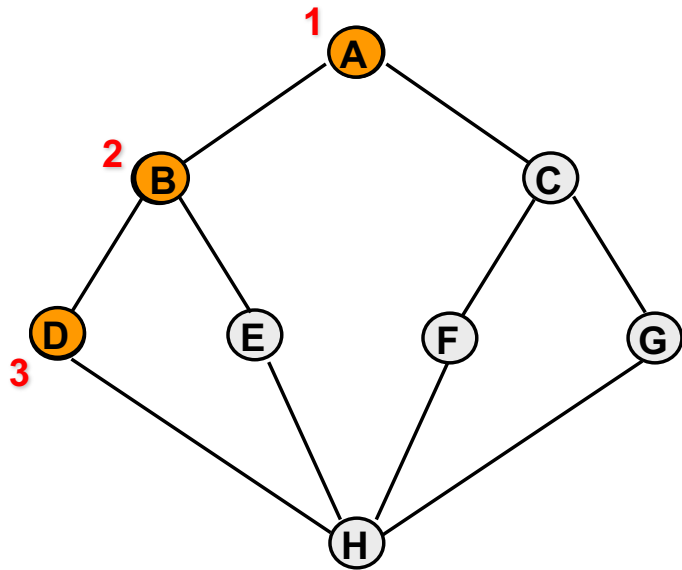


○ não visitado

● visitado

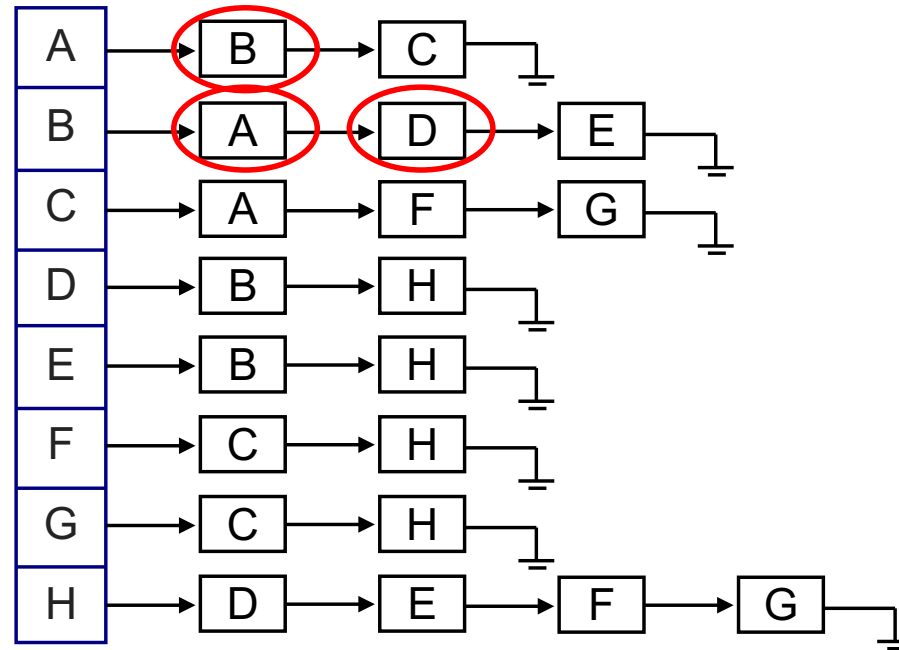


Exemplo

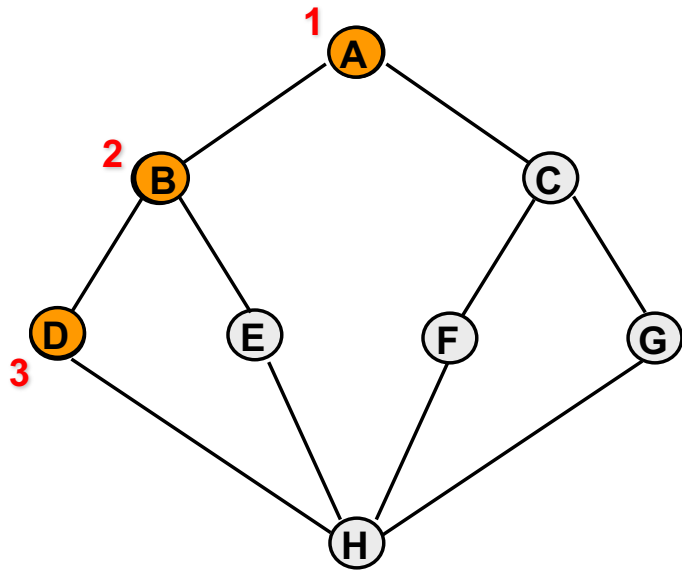


○ não visitado

● visitado

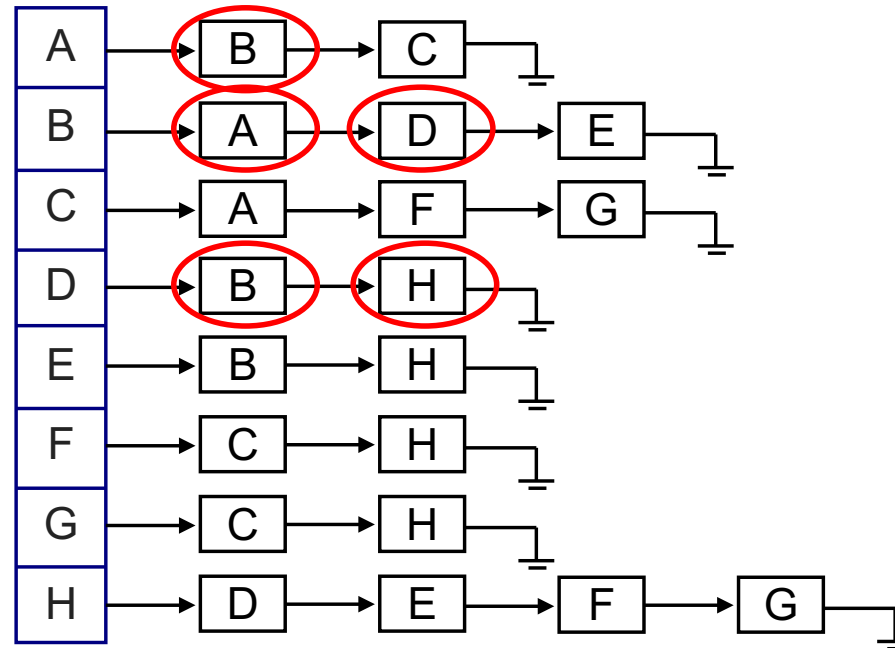


Exemplo

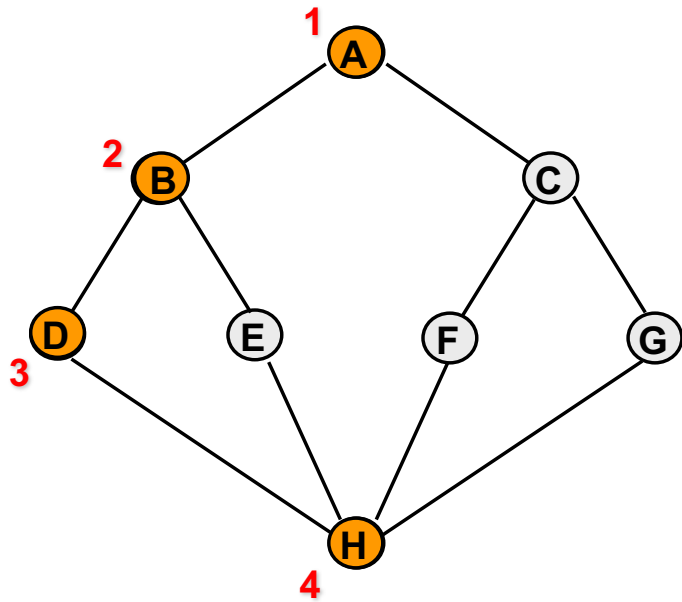


○ não visitado

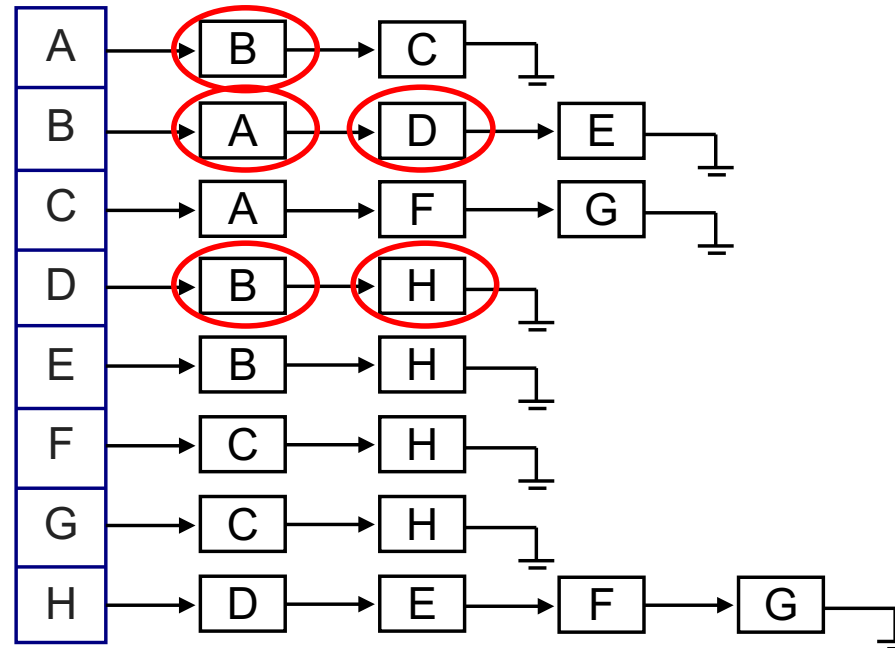
● visitado



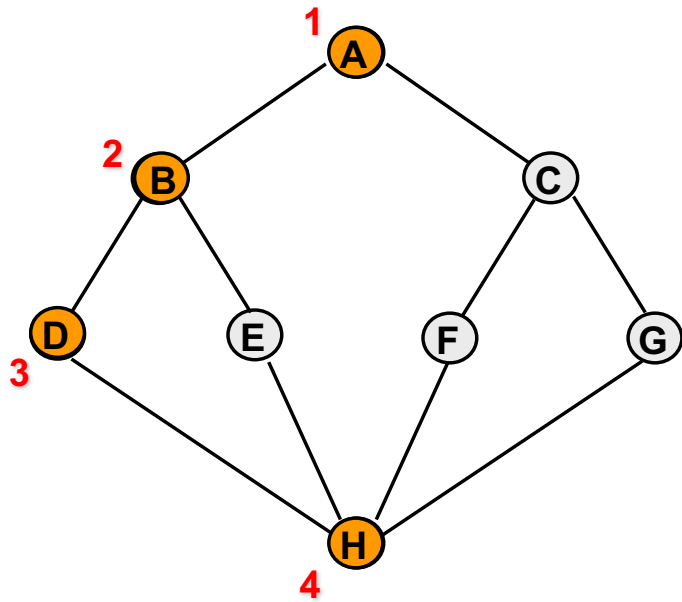
Exemplo



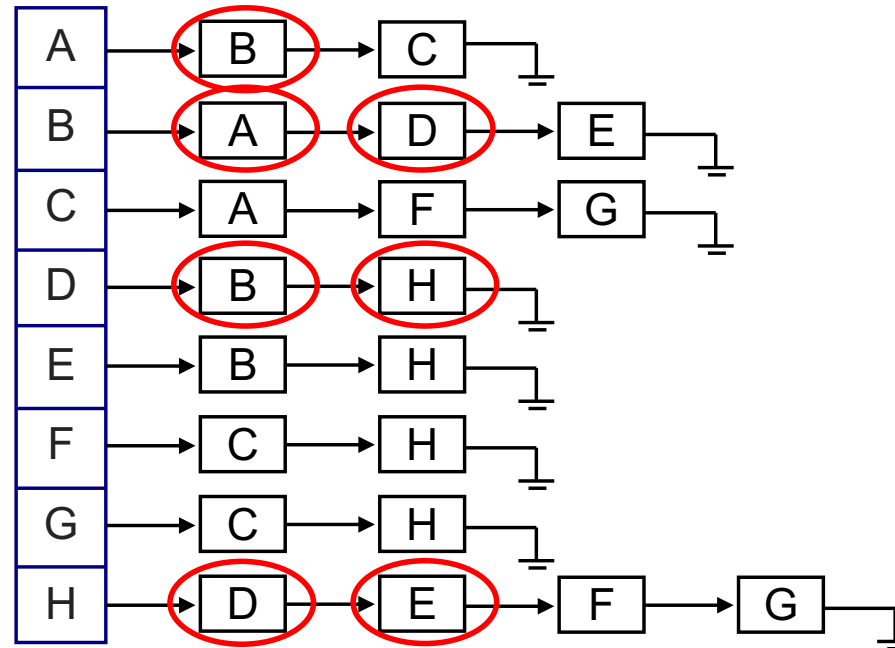
- não visitado
- visitado



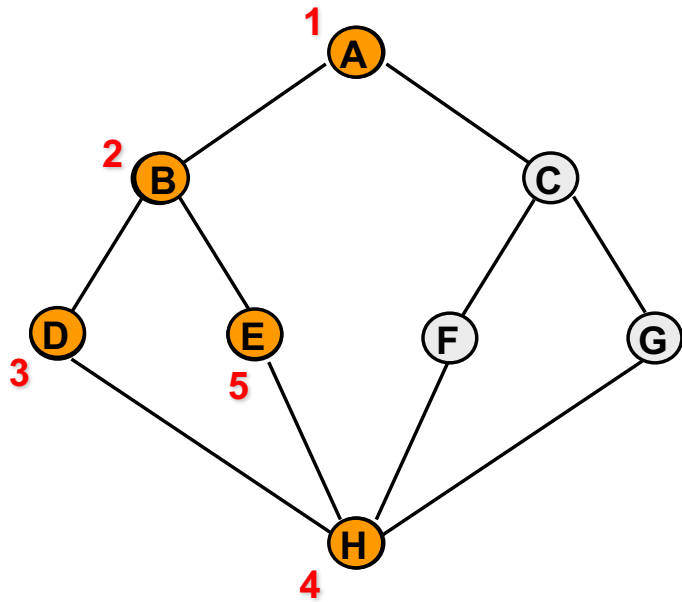
Exemplo



- não visitado
- visitado

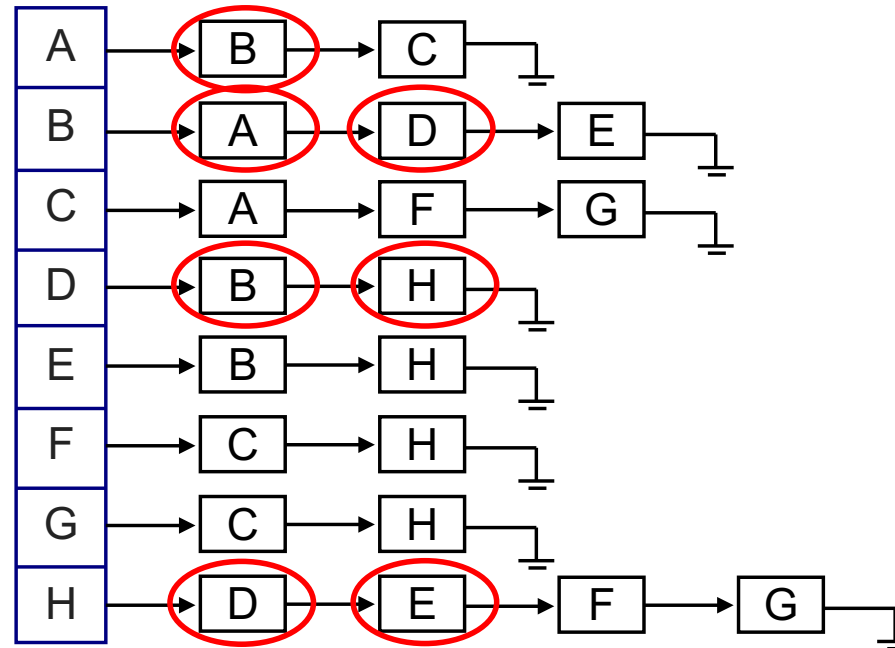


Exemplo

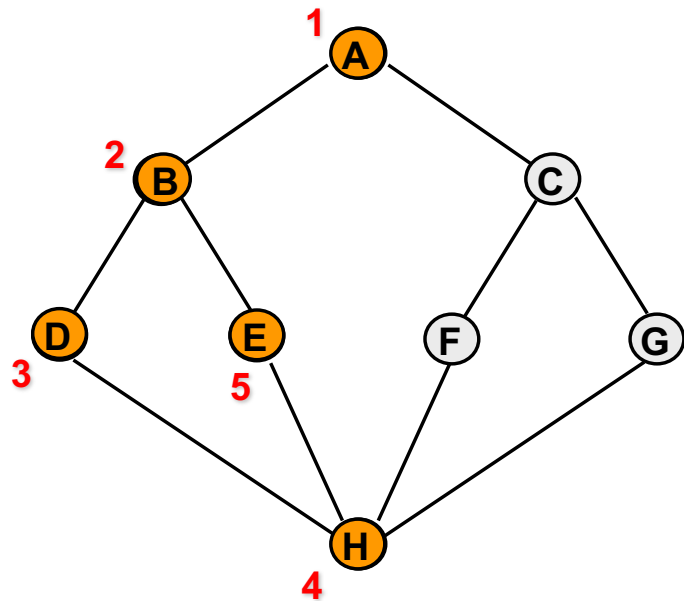


○ não visitado

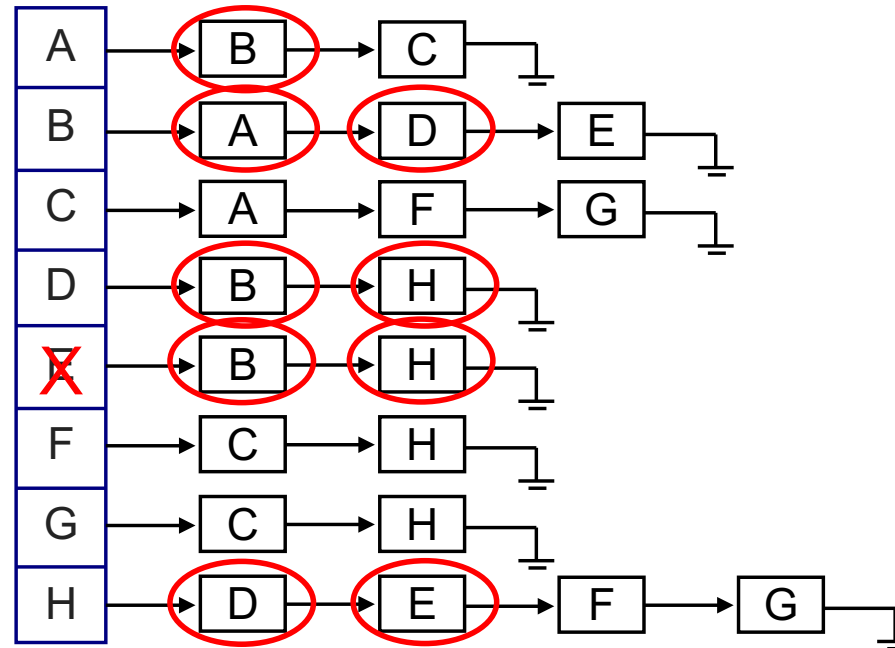
● visitado



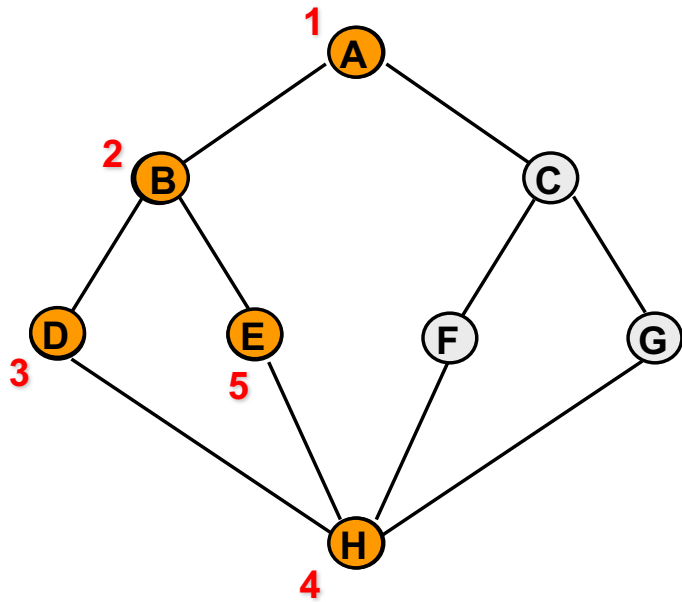
Exemplo



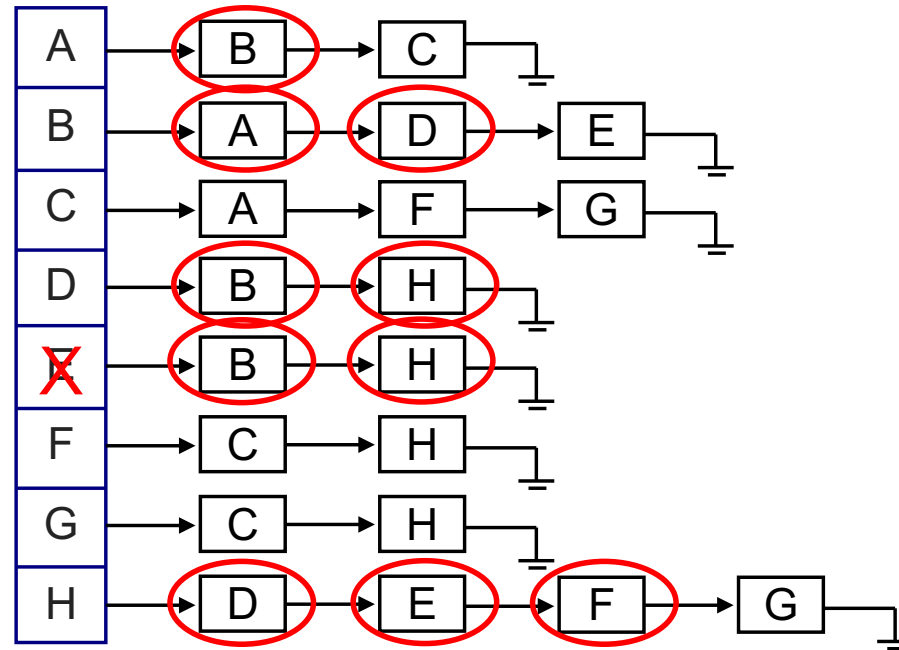
- não visitado
- visitado



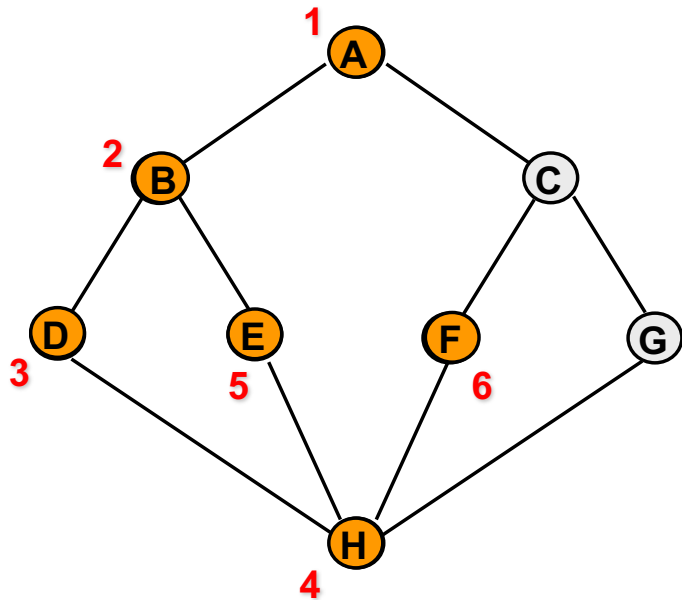
Exemplo



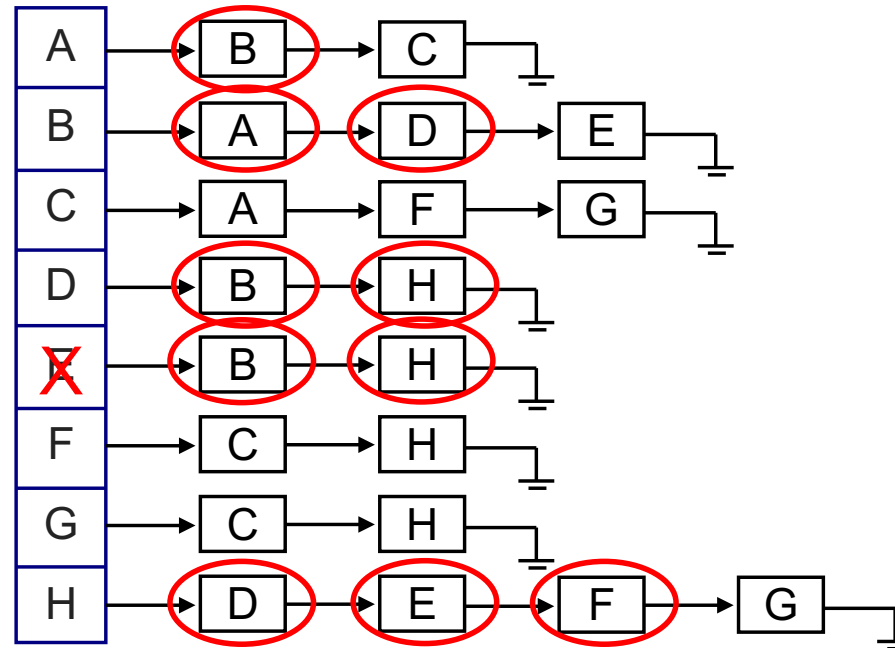
- não visitado
- visitado



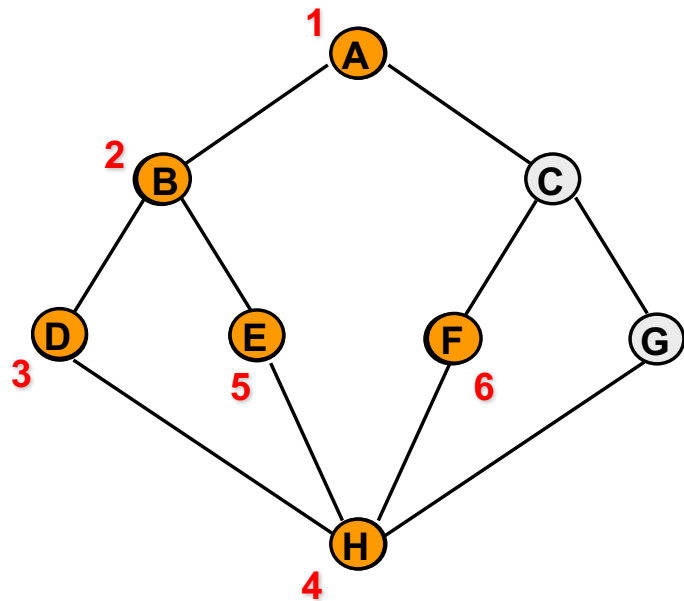
Exemplo



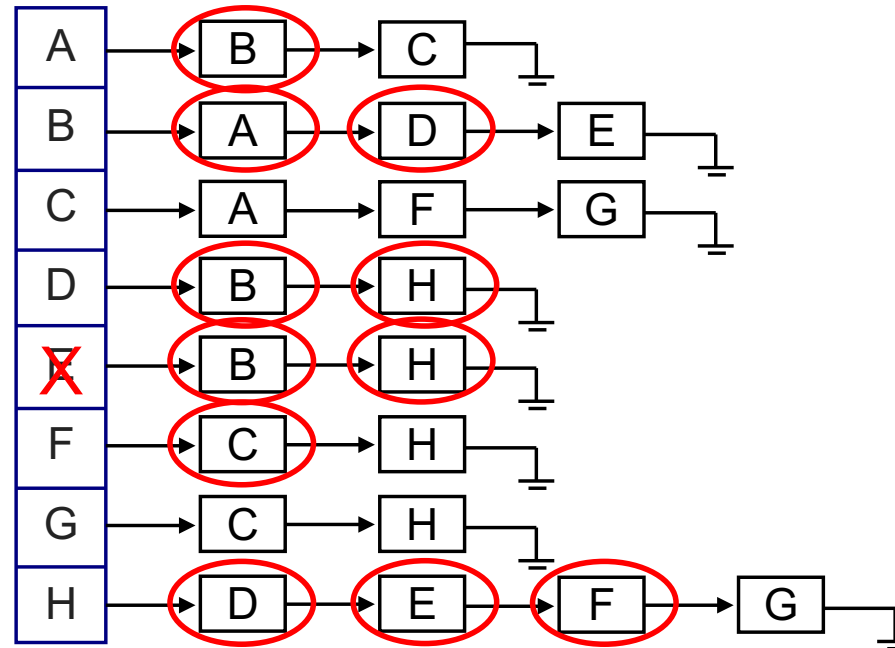
- não visitado
- visitado



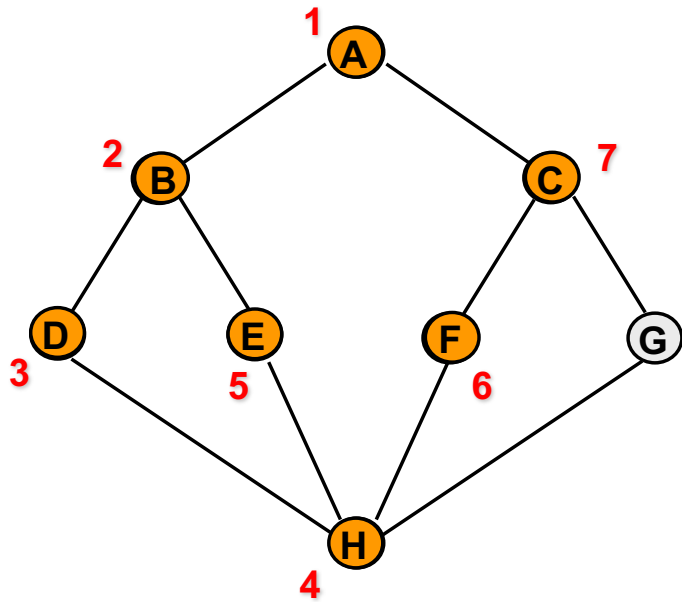
Exemplo



- não visitado
- visitado

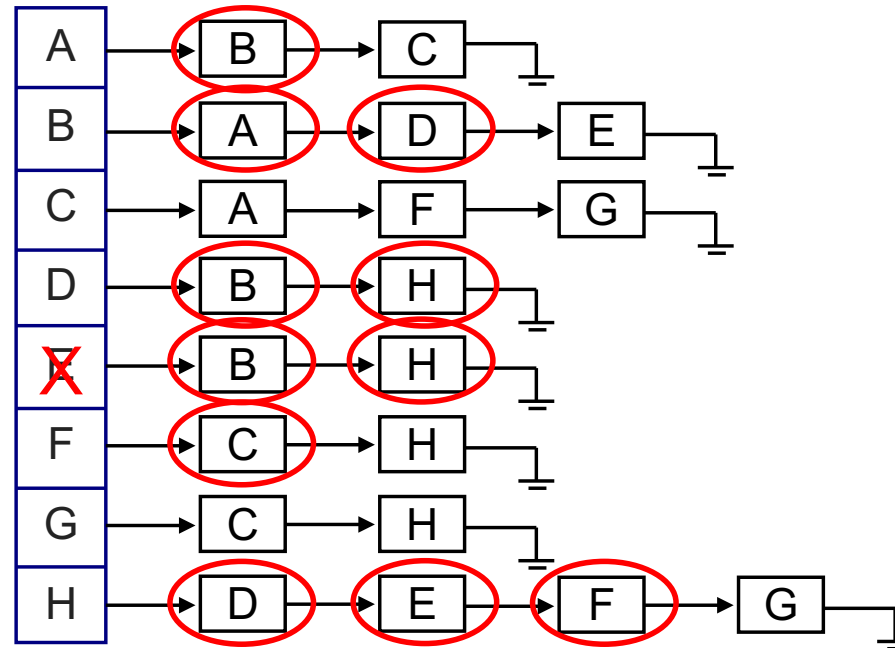


Exemplo

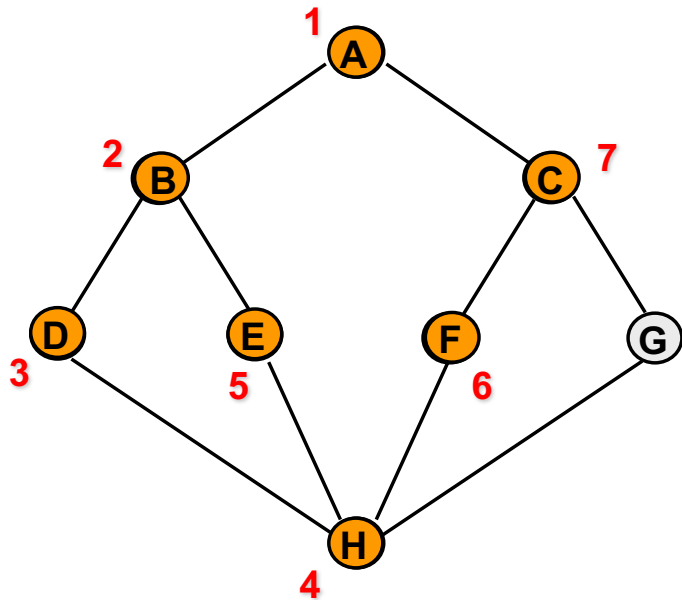


○ não visitado

● visitado

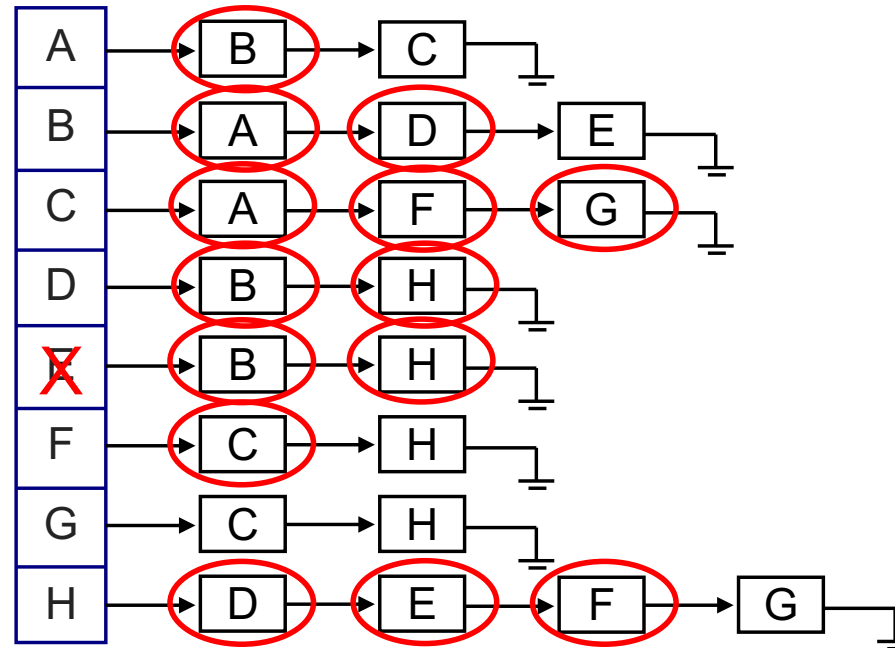


Exemplo

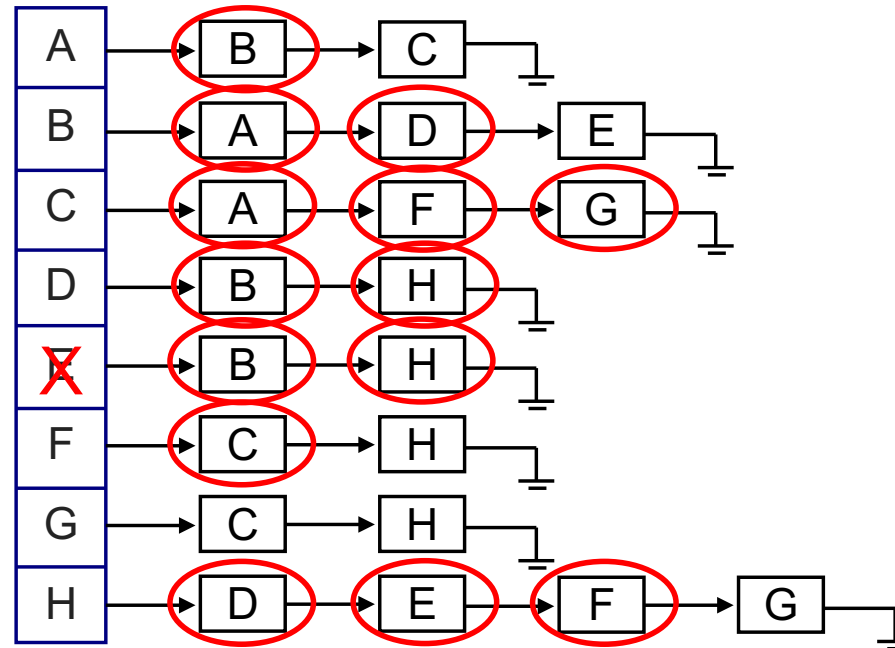
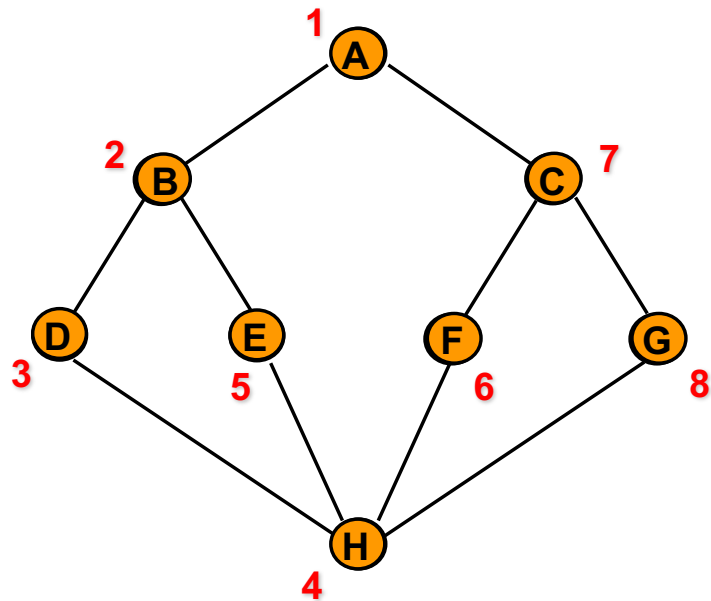


○ não visitado

● visitado



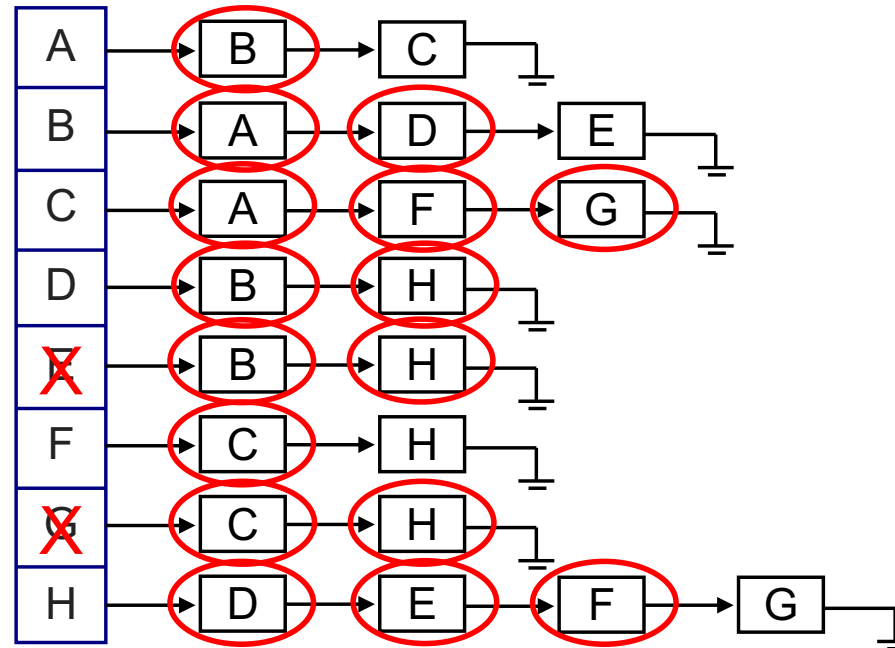
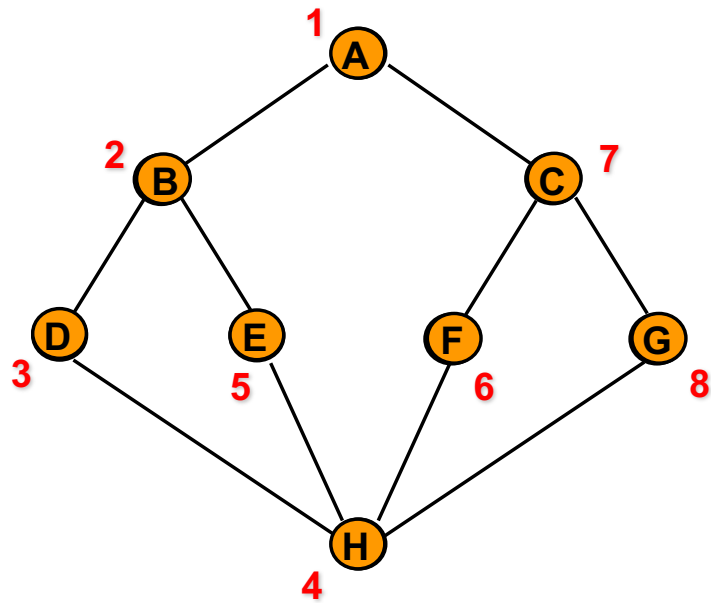
Exemplo



○ não visitado

● visitado

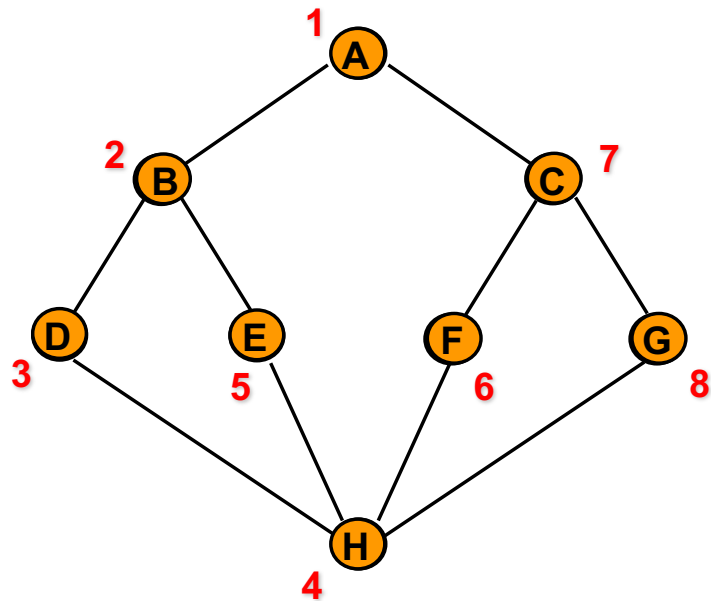
Exemplo



○ não visitado

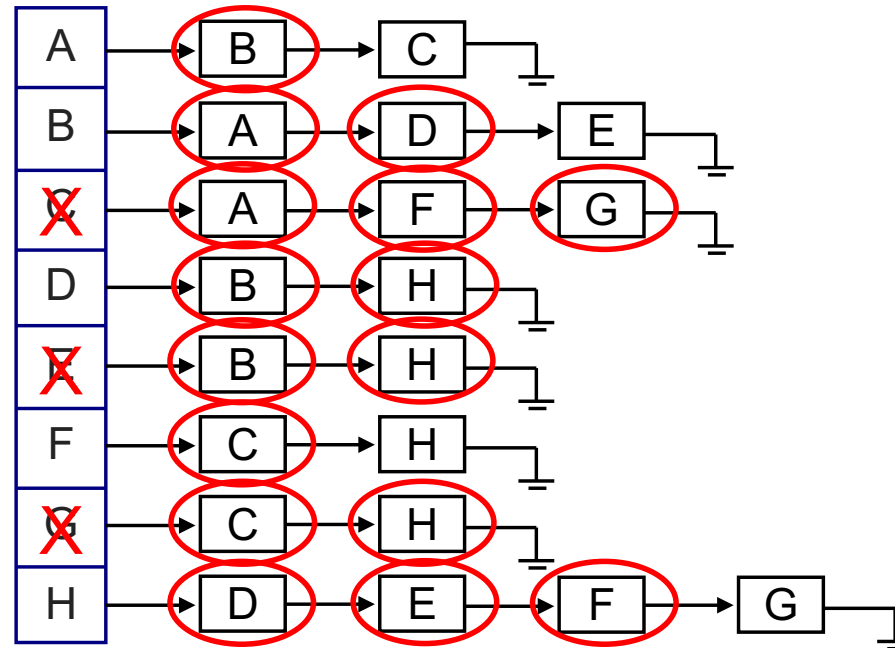
● visitado

Exemplo

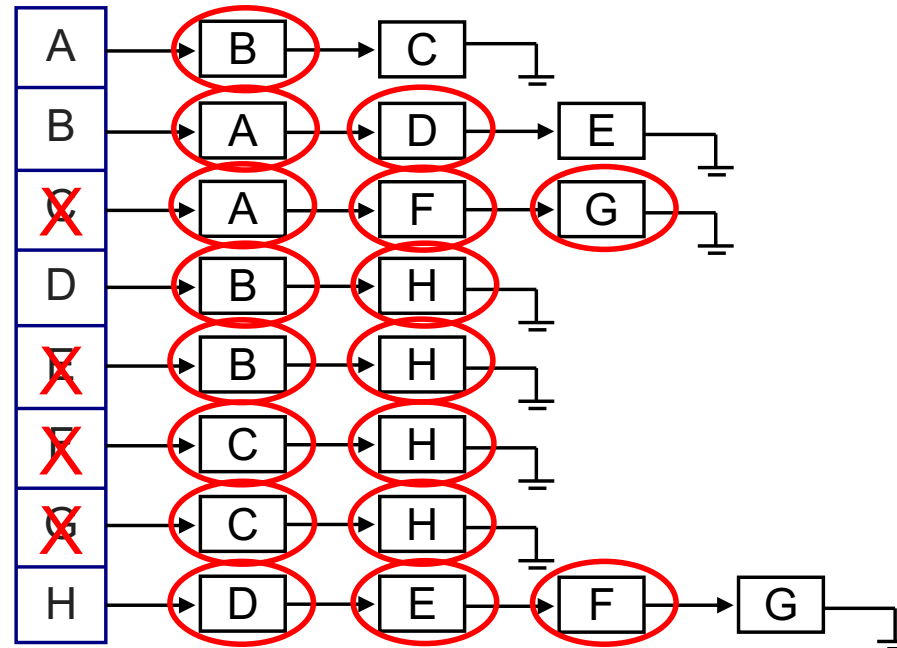
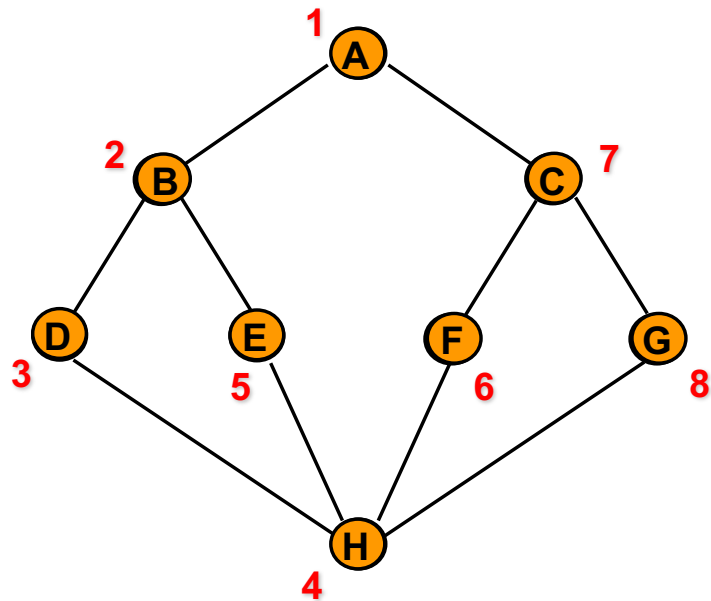


○ não visitado

● visitado



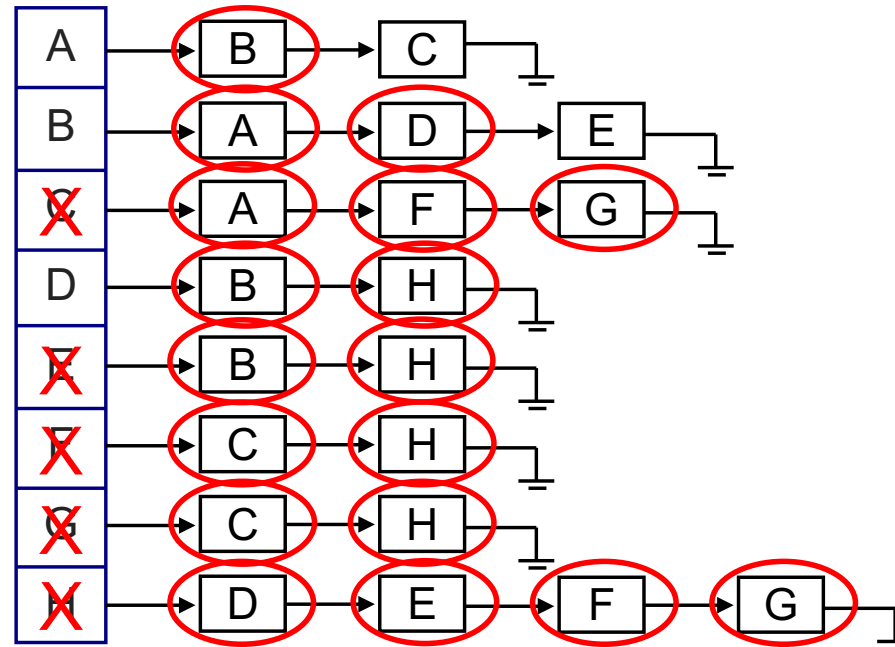
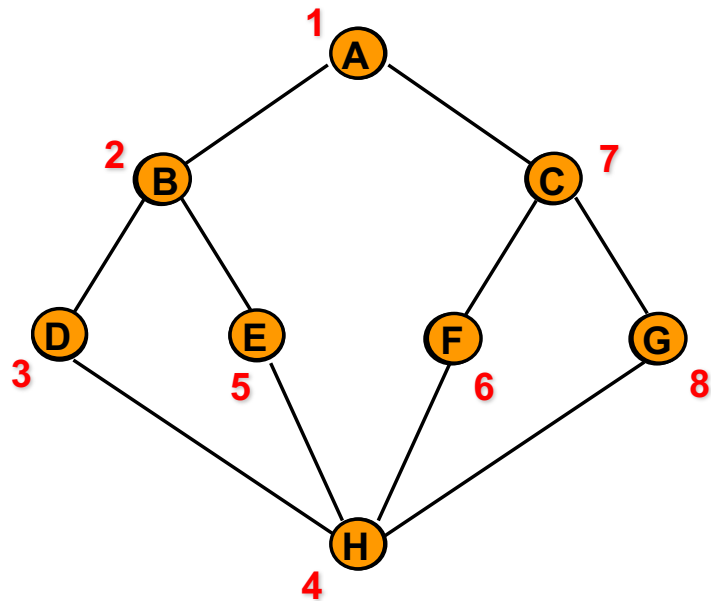
Exemplo



○ não visitado

● visitado

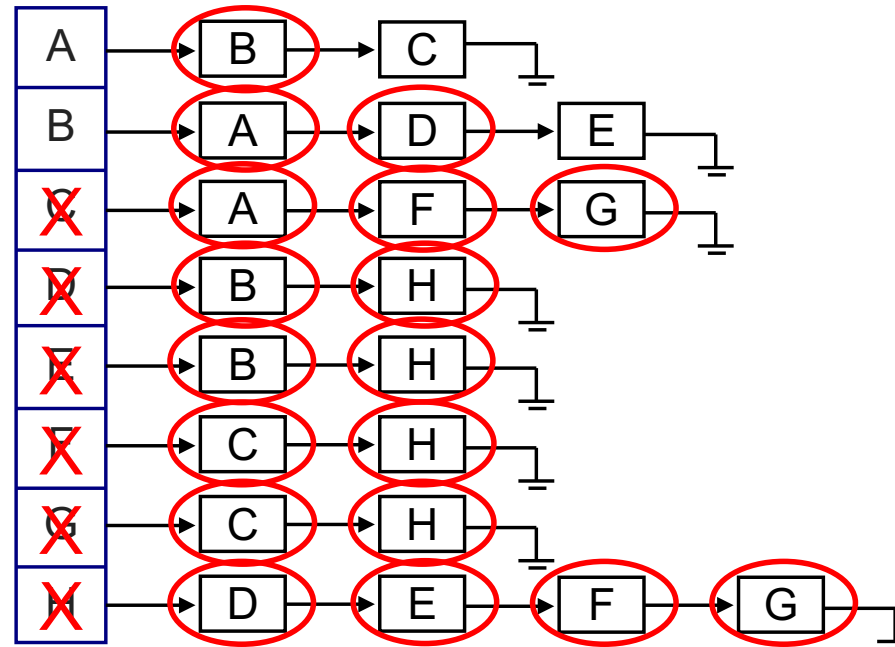
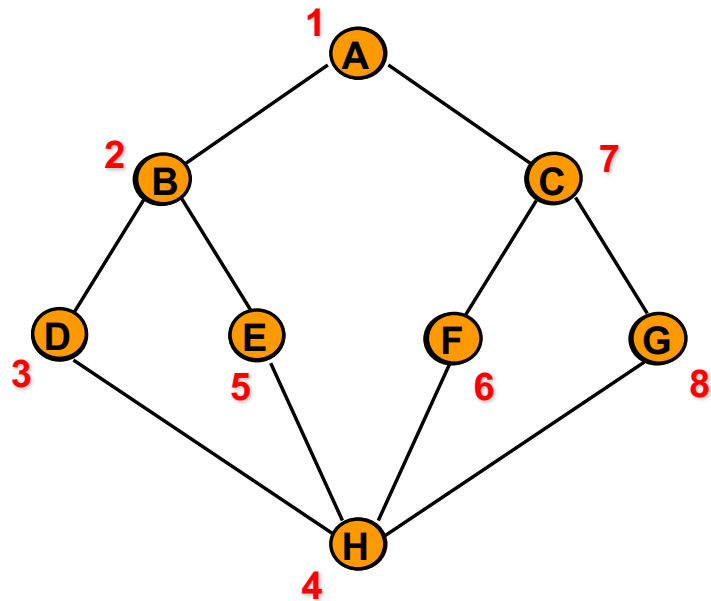
Exemplo



○ não visitado

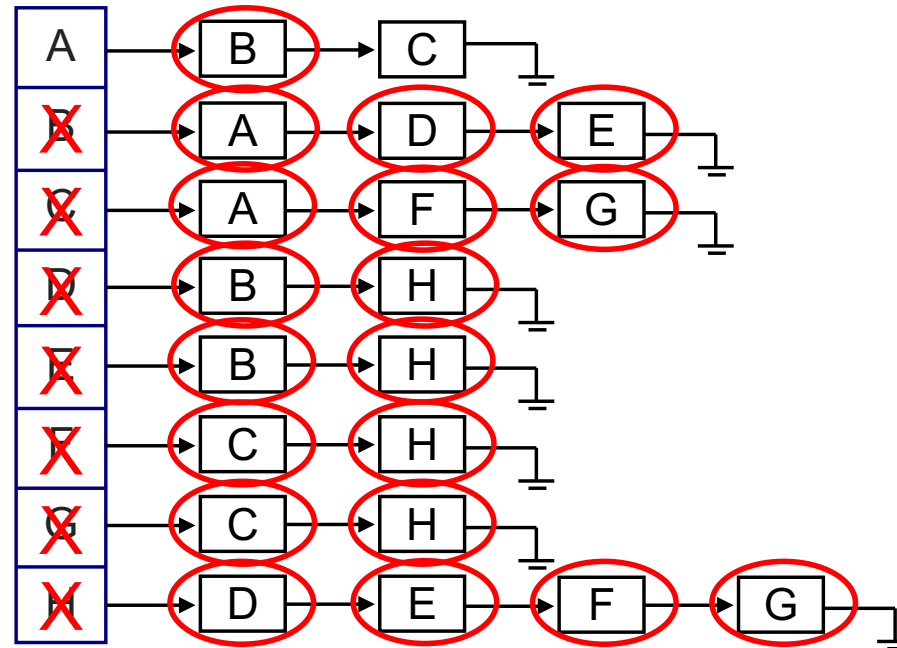
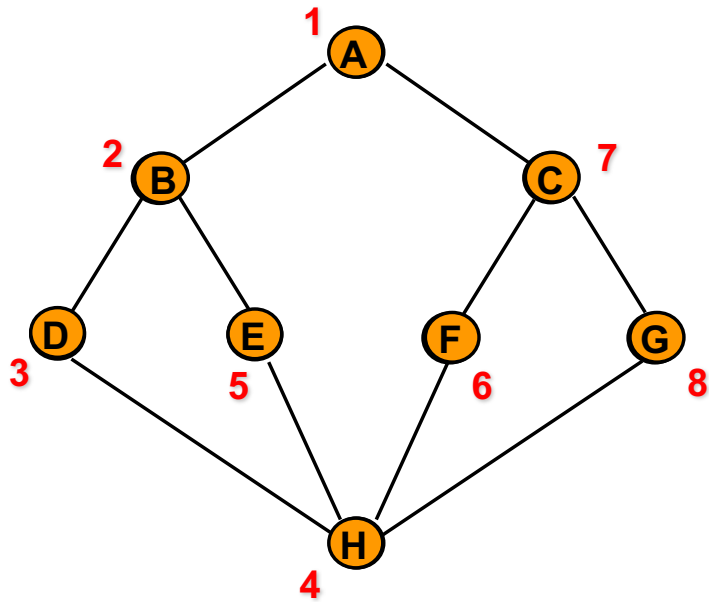
● visitado

Exemplo



- não visitado
- visitado

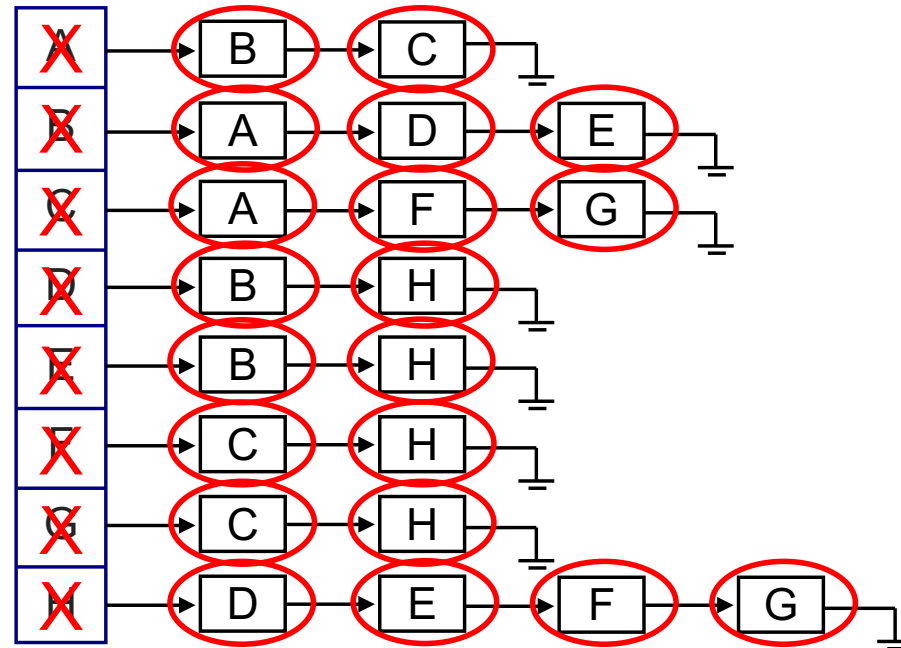
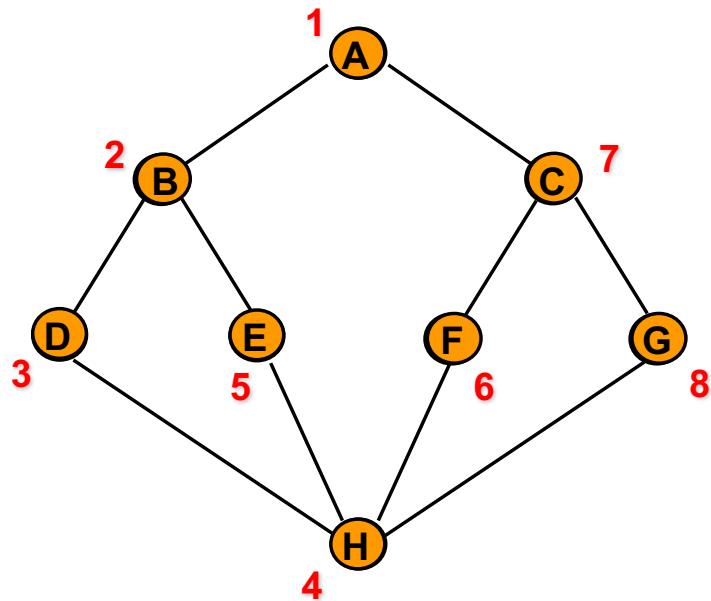
Exemplo



○ não visitado

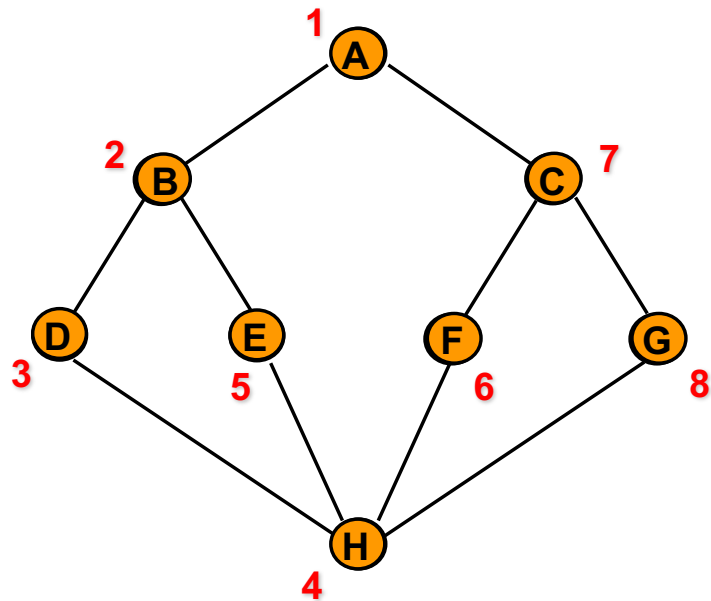
● visitado

Exemplo

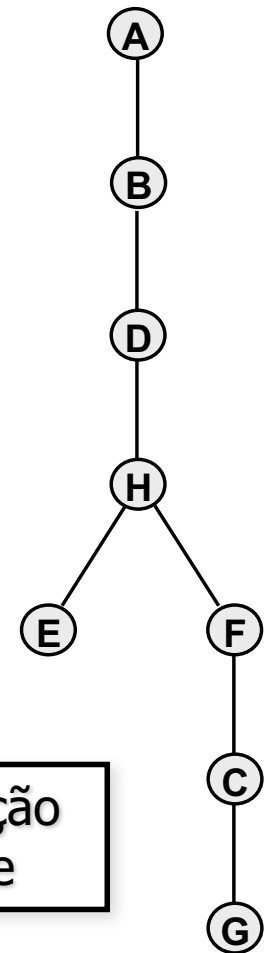
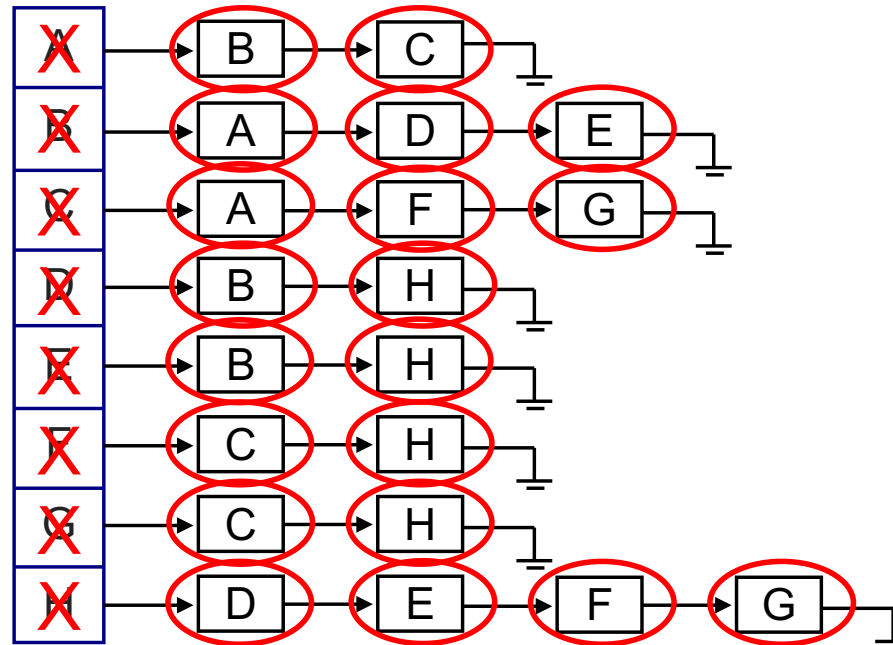


- não visitado
- visitado

Exemplo



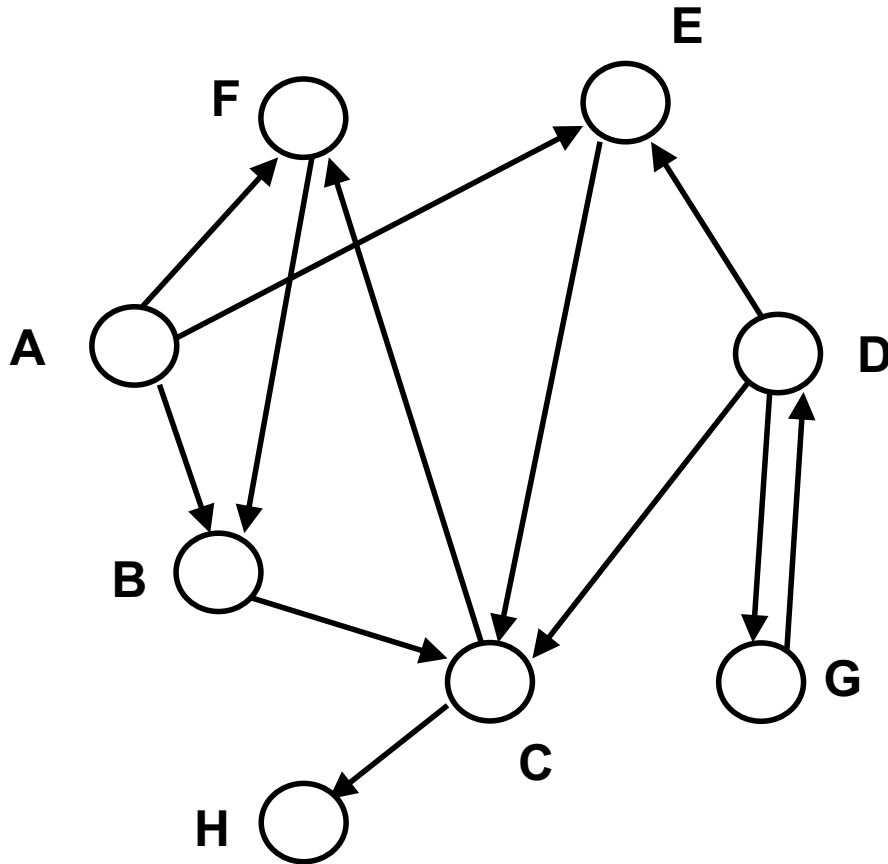
- não visitado
- visitado



Árvore de exploração em profundidade

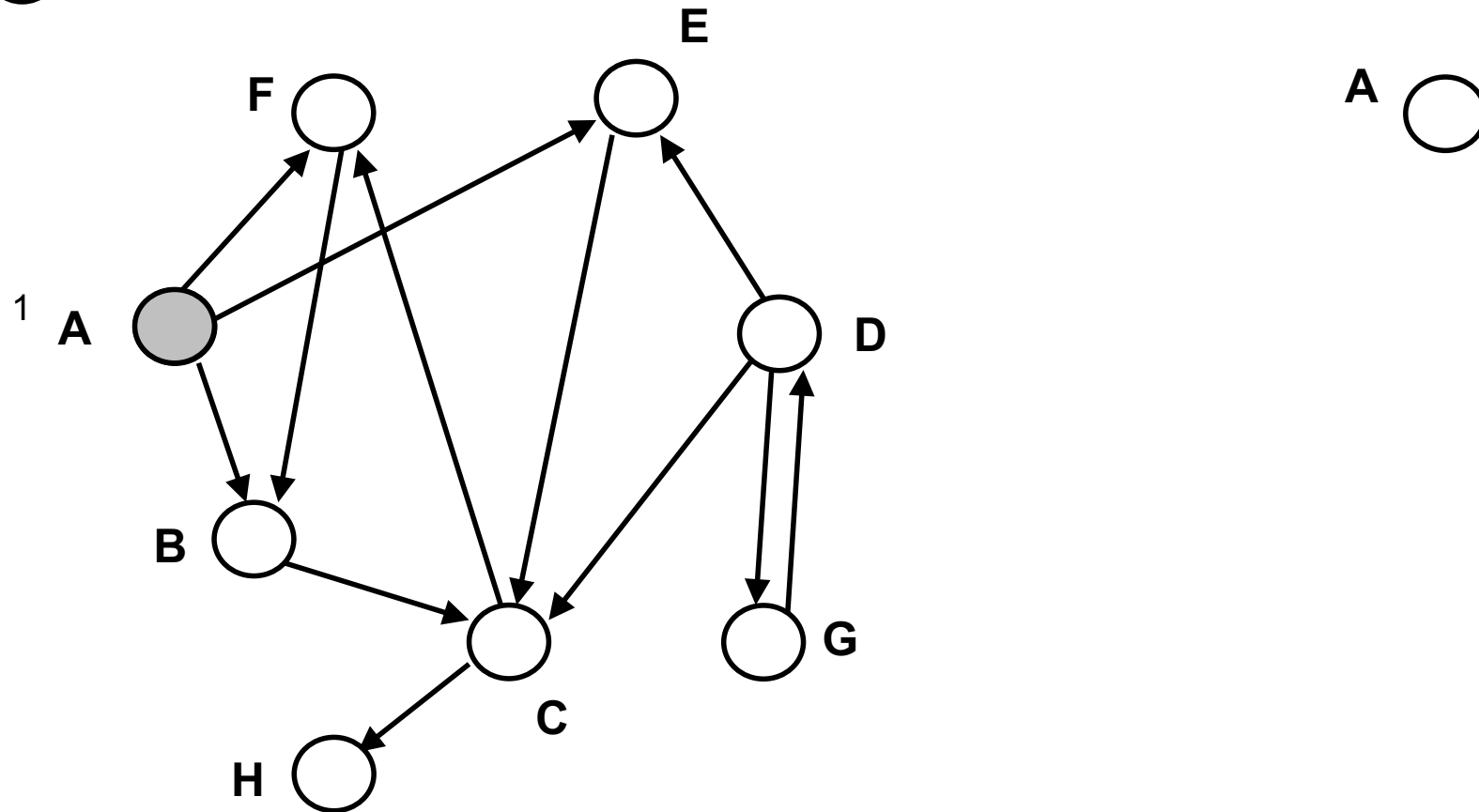
Exemplo

- *Não visitado*
- ◐ *Em exploração*
- *Terminado*



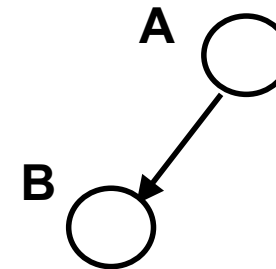
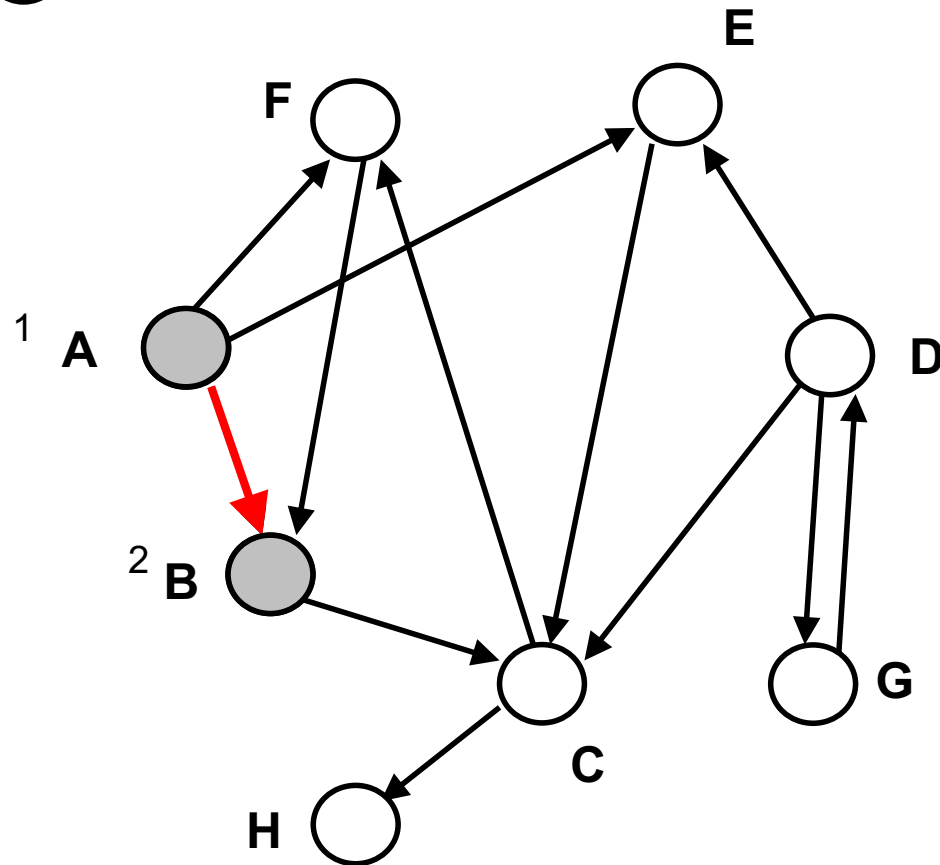
Exemplo

- *Não visitado*
- ◐ *Em exploração*
- *Terminado*



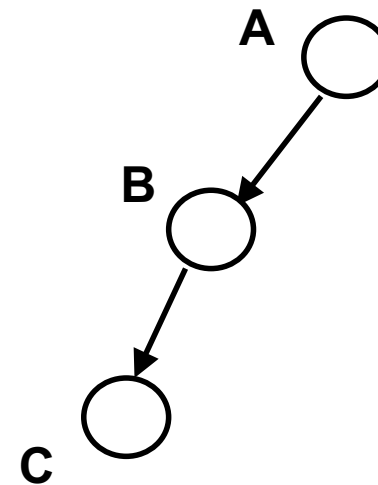
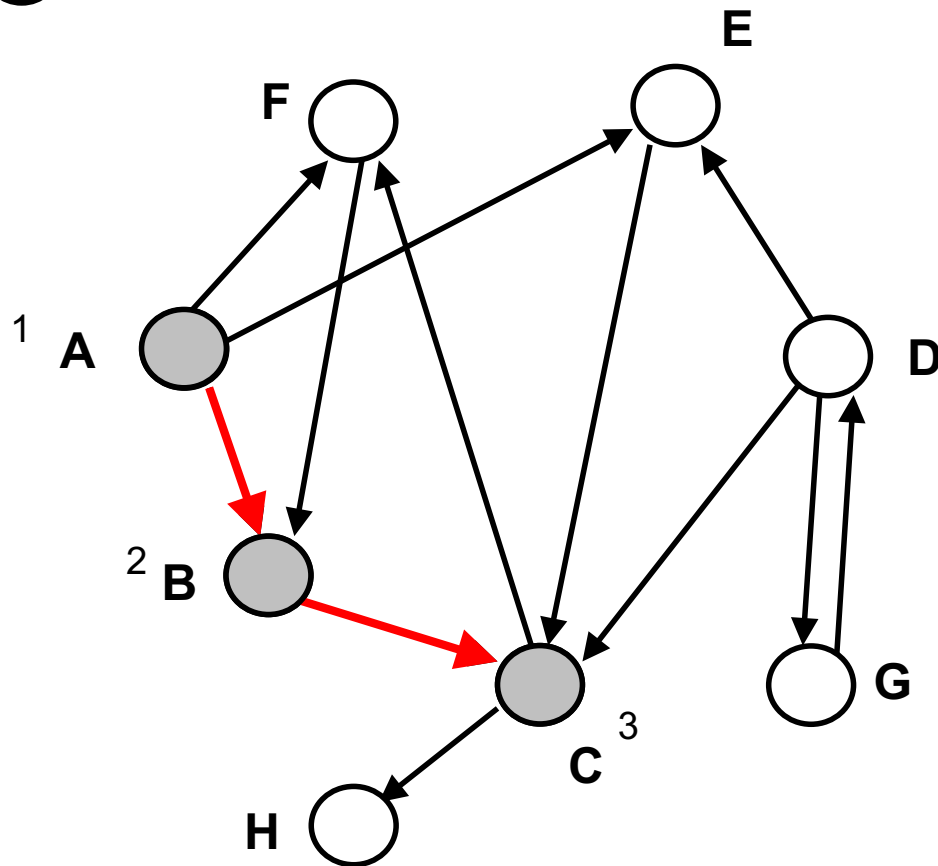
Exemplo

- *Não visitado*
- ◐ *Em exploração*
- *Terminado*



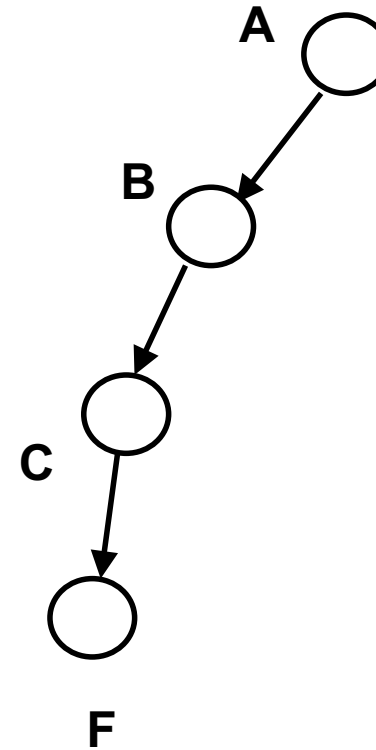
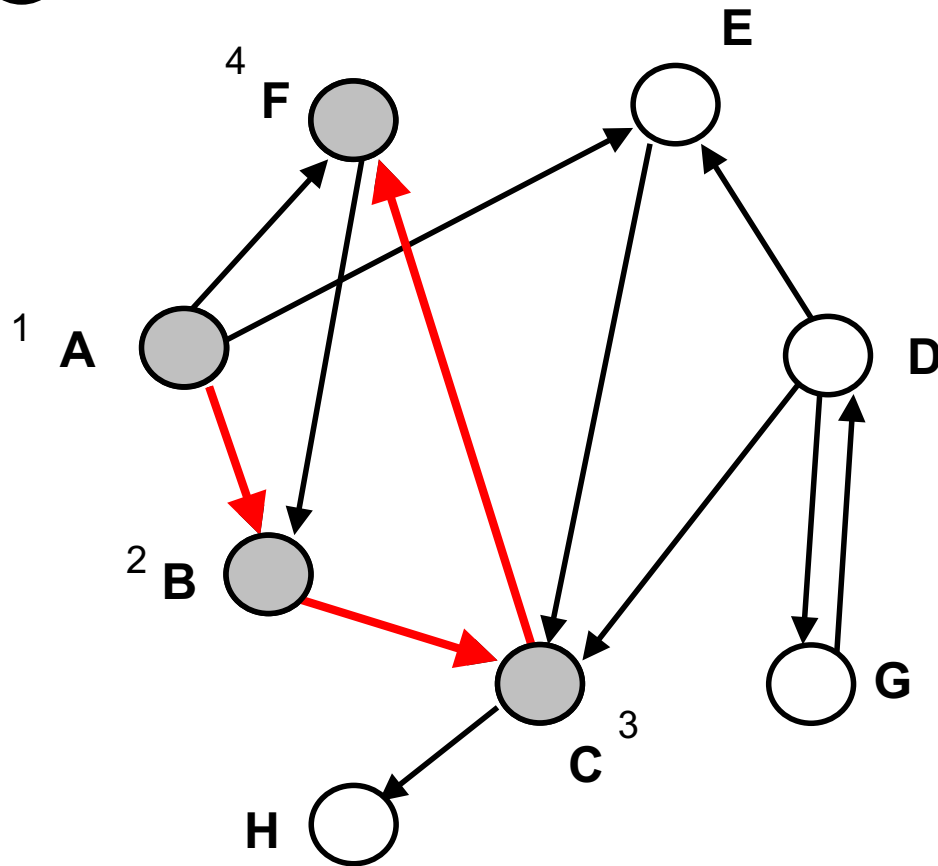
Exemplo

- Não visitado
- Em exploração
- Terminado



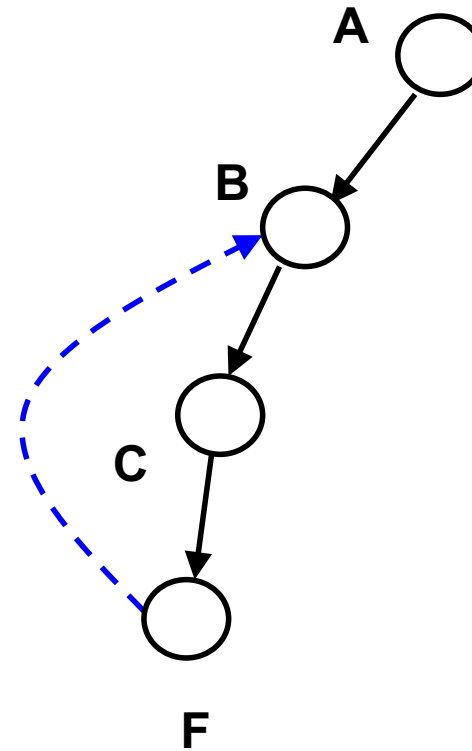
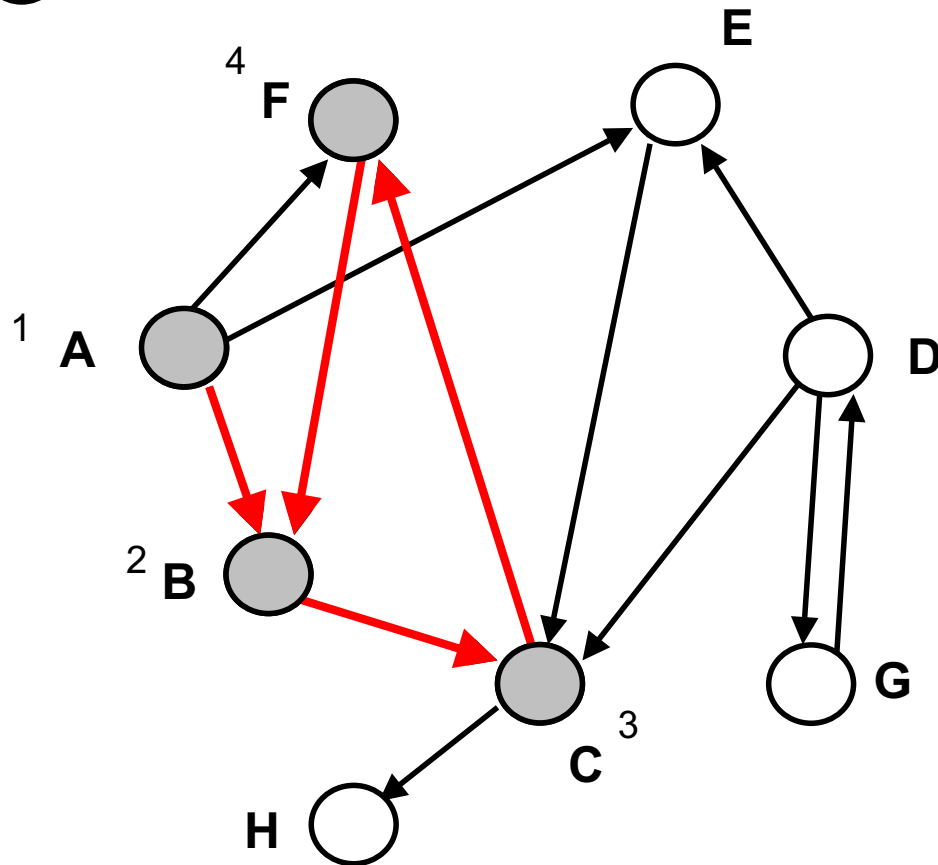
Exemplo

- Não visitado
- Em exploração
- Terminado



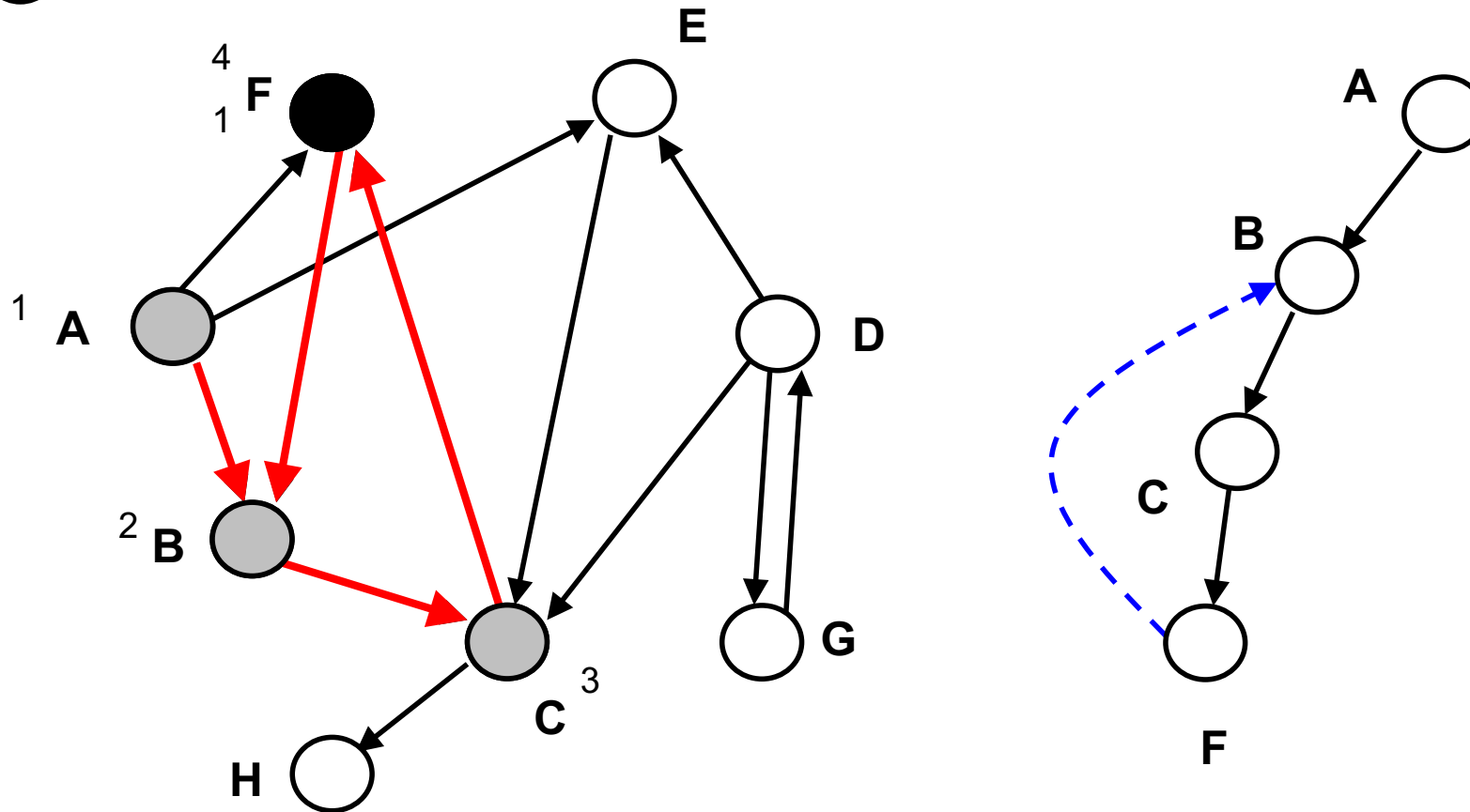
Exemplo

- Não visitado
- Em exploração
- Terminado



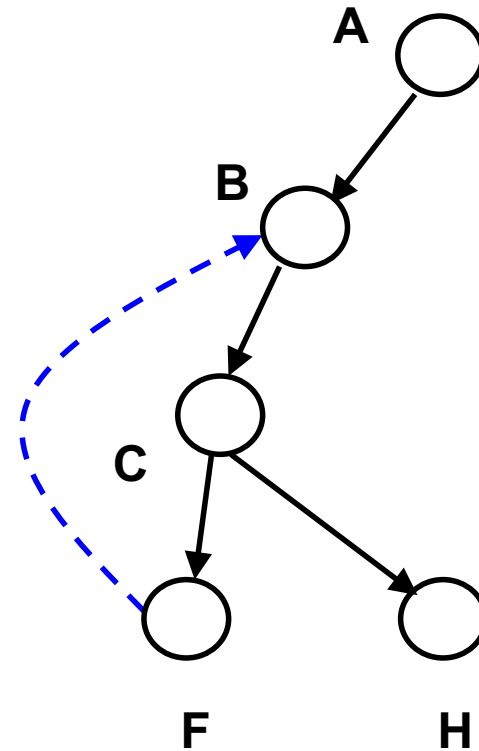
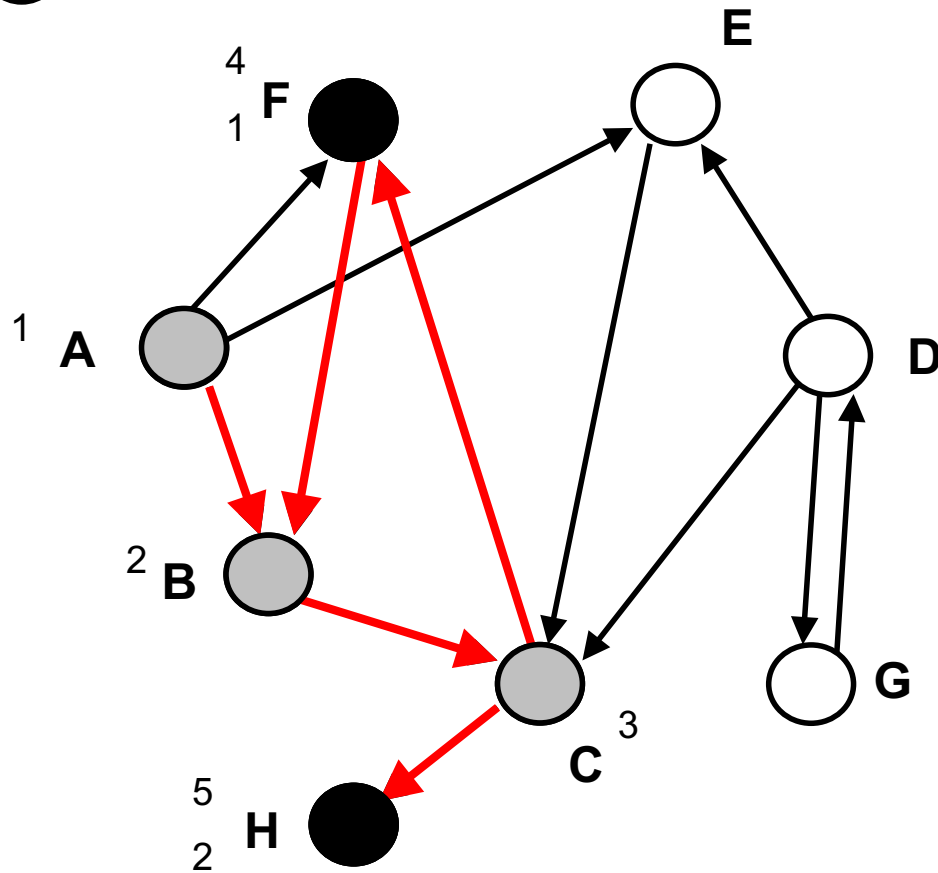
Exemplo

- Não visitado
- Em exploração
- Terminado



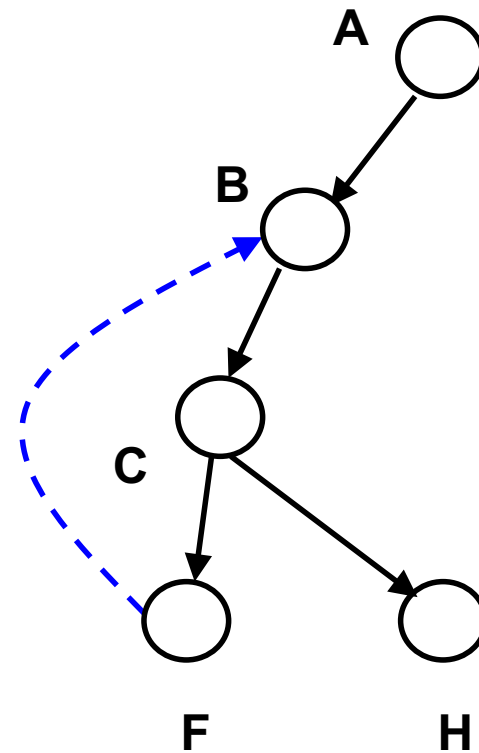
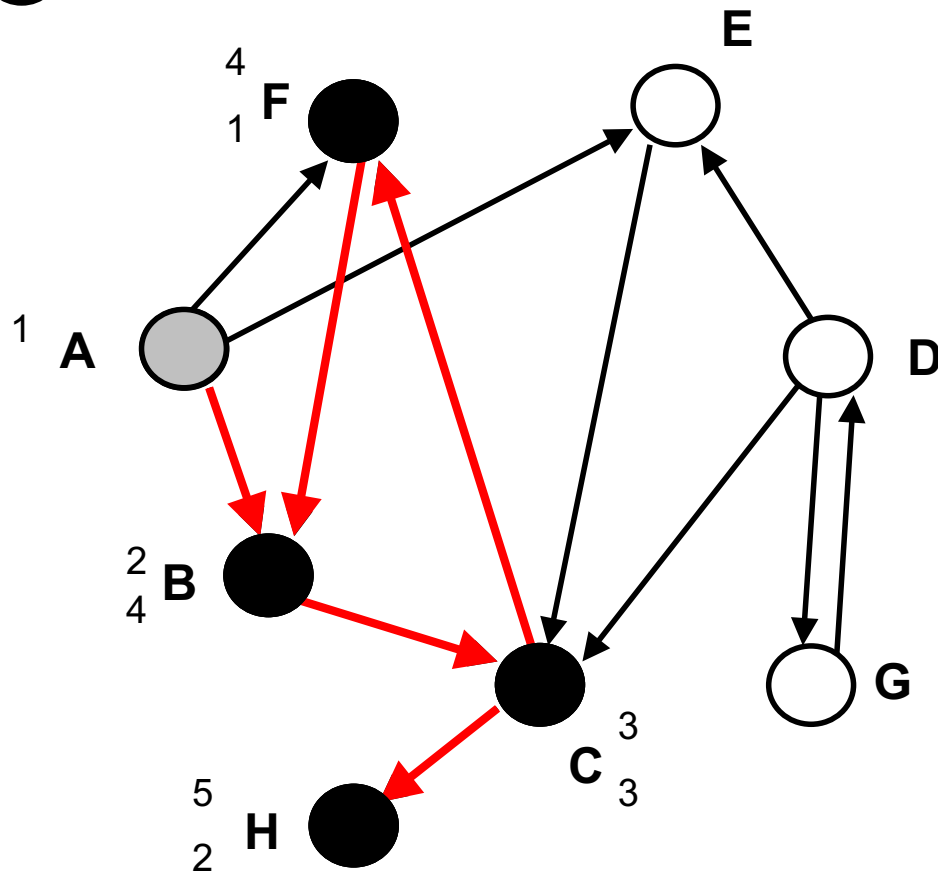
Exemplo

- Não visitado
- Em exploração
- Terminado



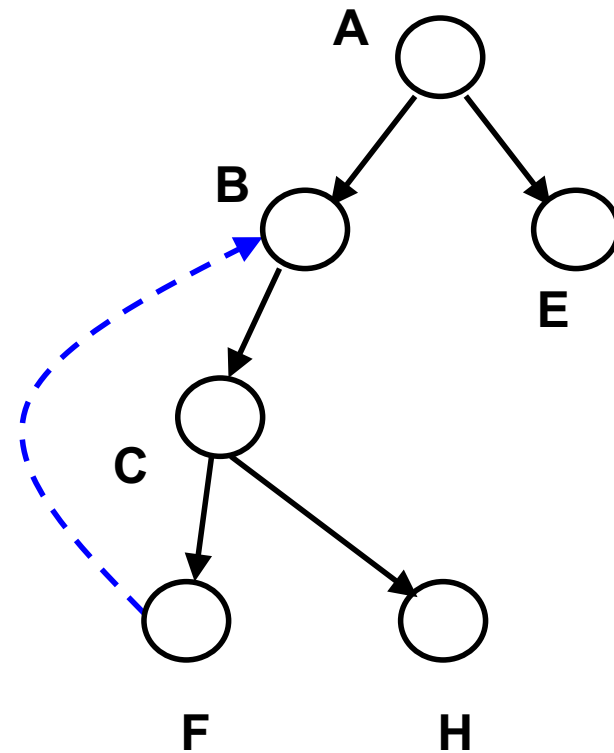
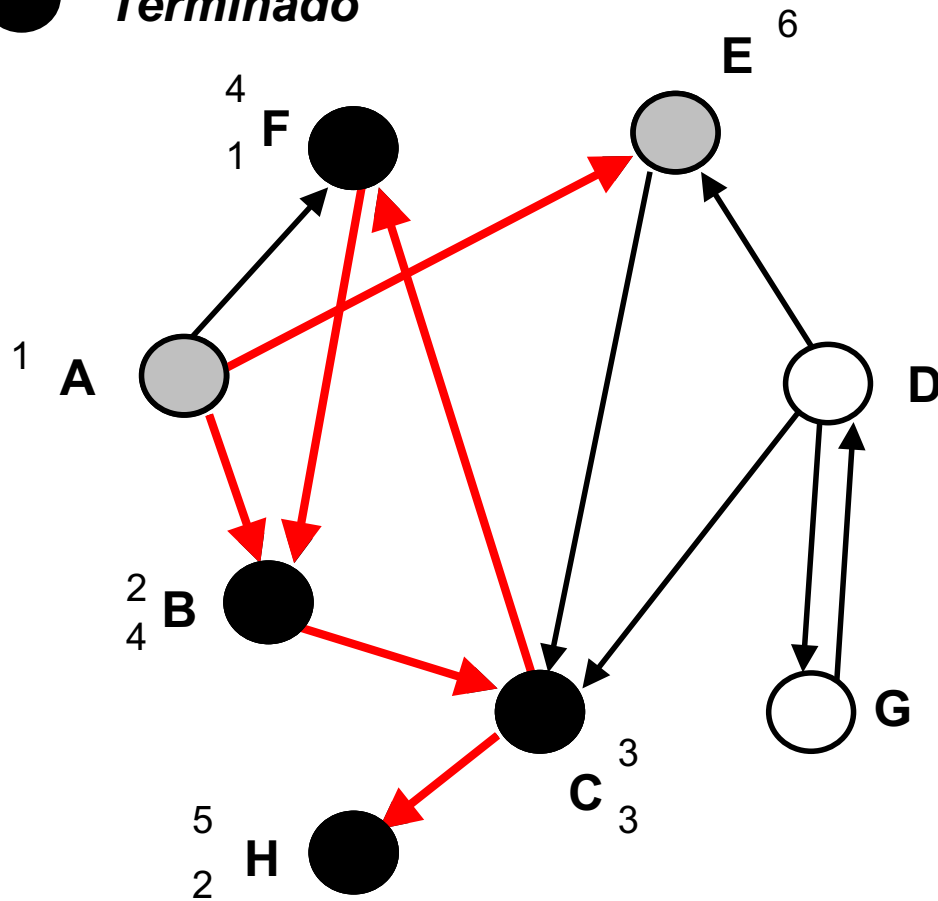
Exemplo

- Não visitado
- ◐ Em exploração
- Terminado



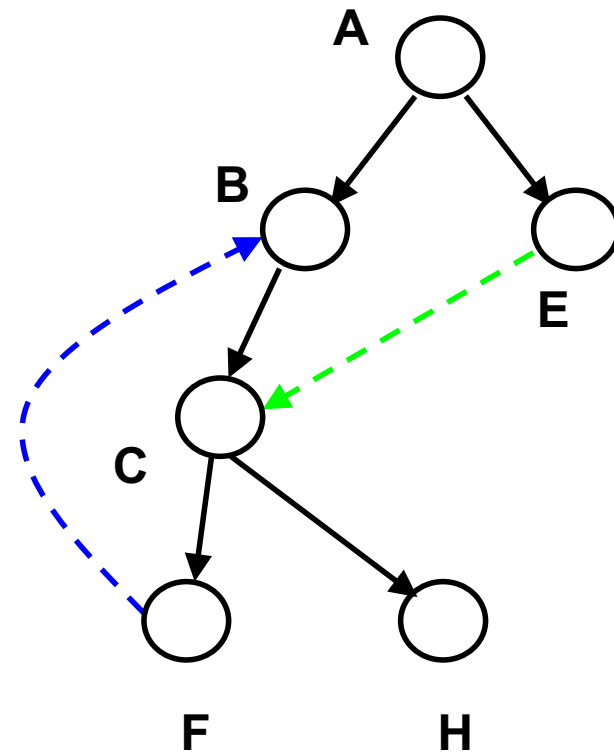
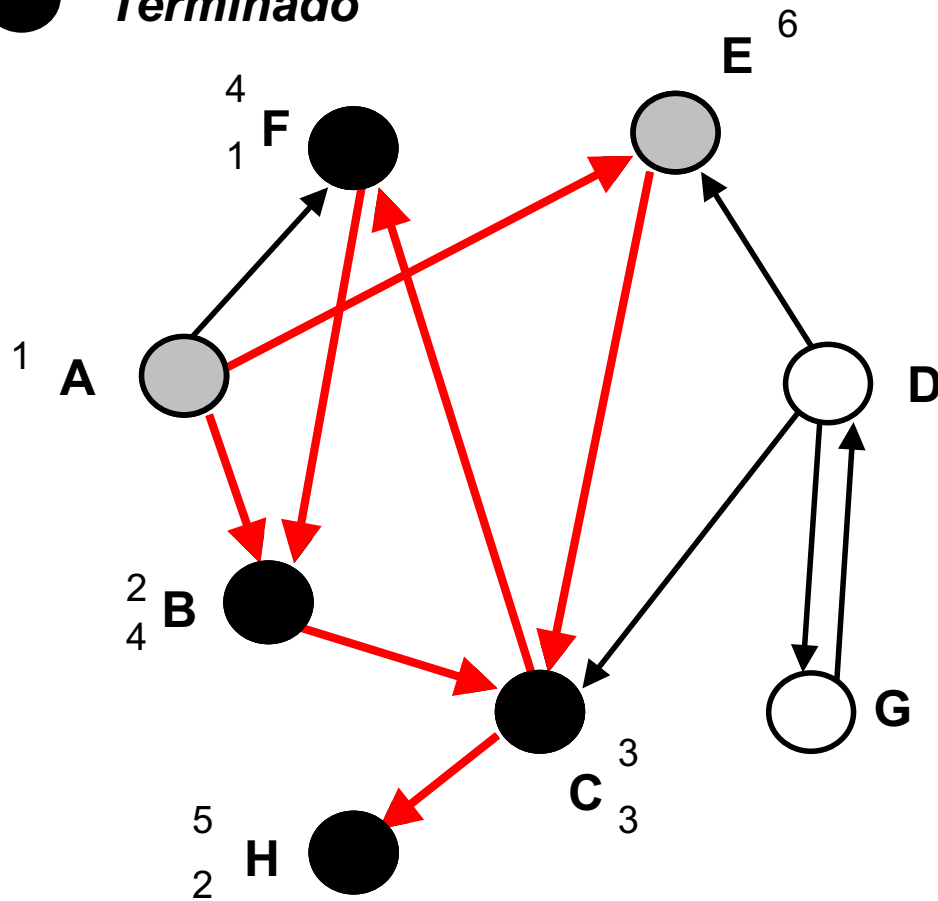
Exemplo

- Não visitado
- Em exploração
- Terminado



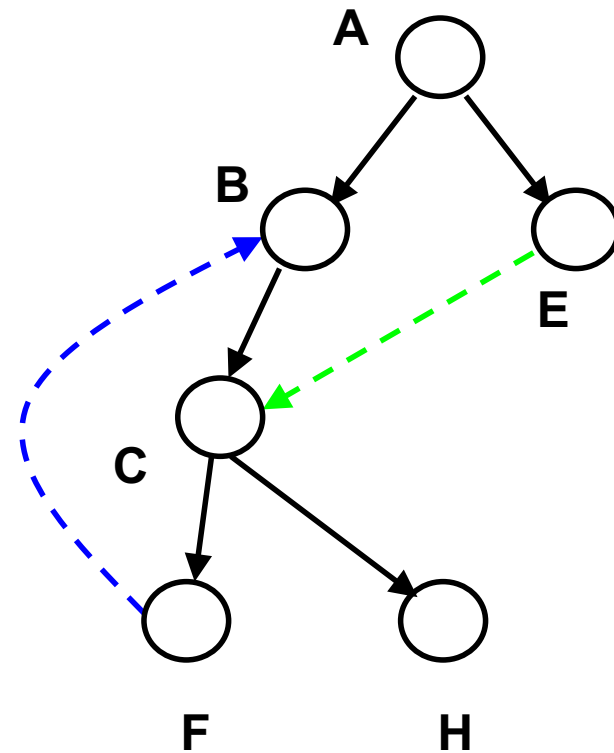
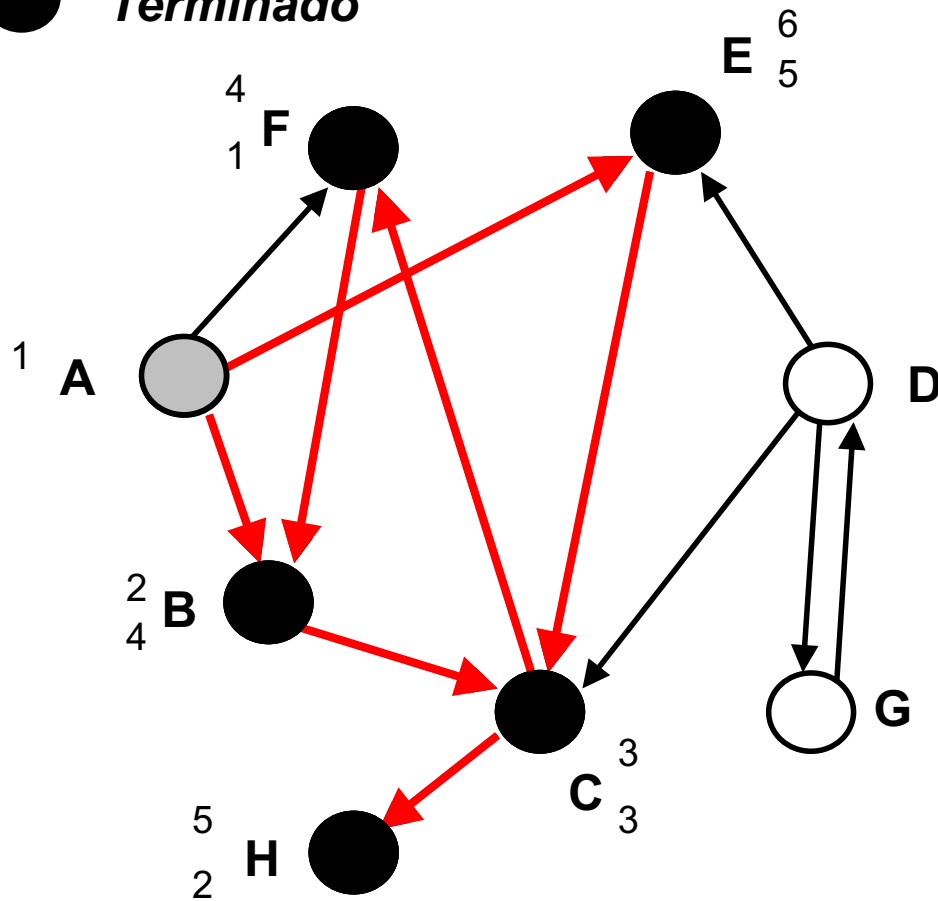
Exemplo

- Não visitado
- Em exploração
- Terminado



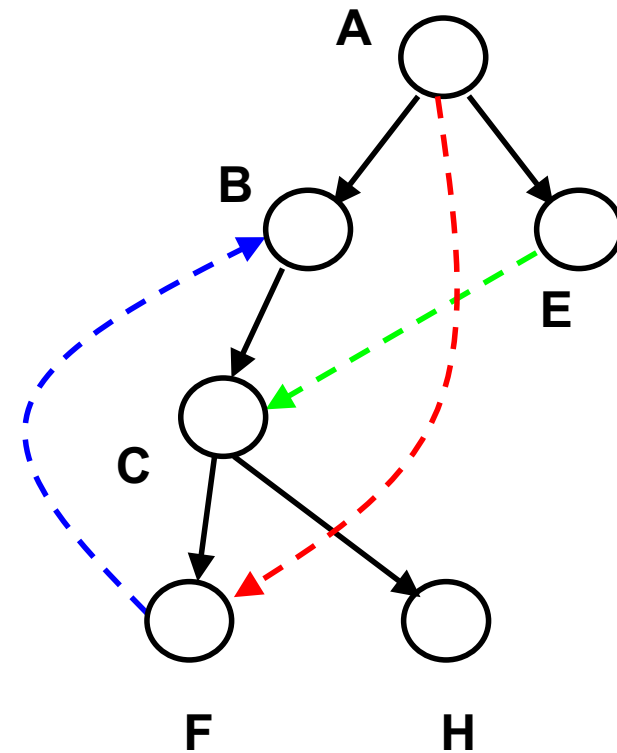
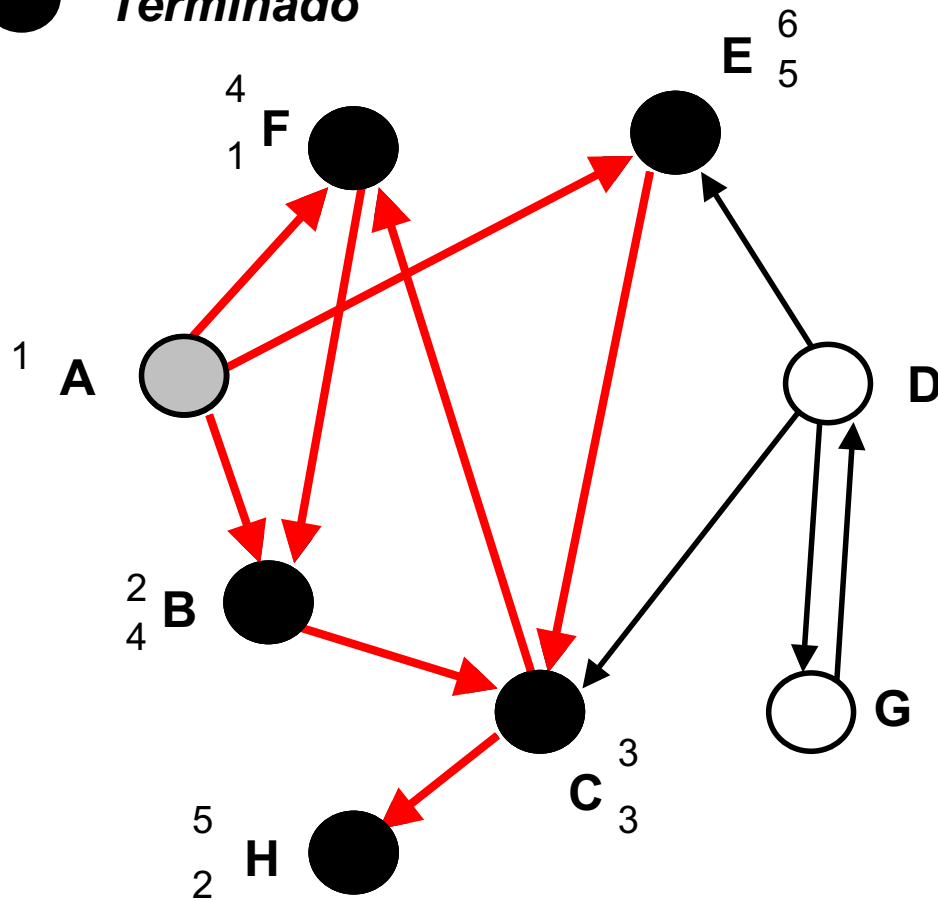
Exemplo

- Não visitado
- Em exploração
- Terminado



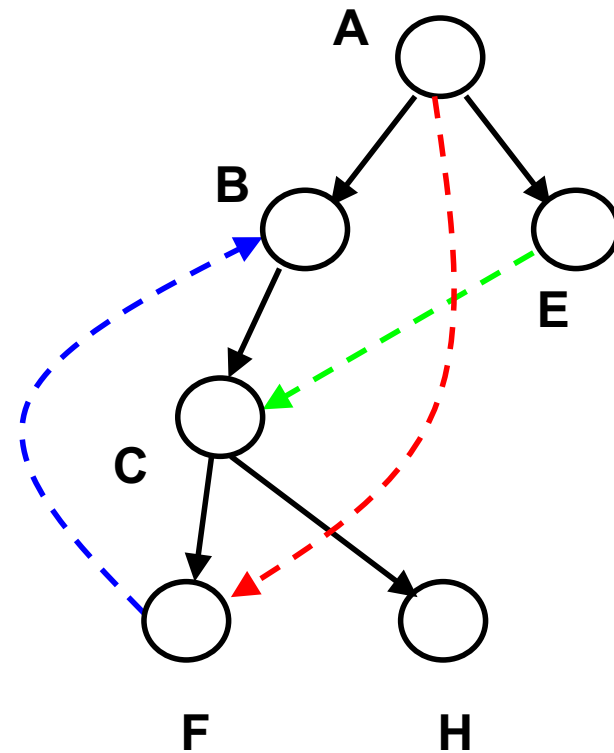
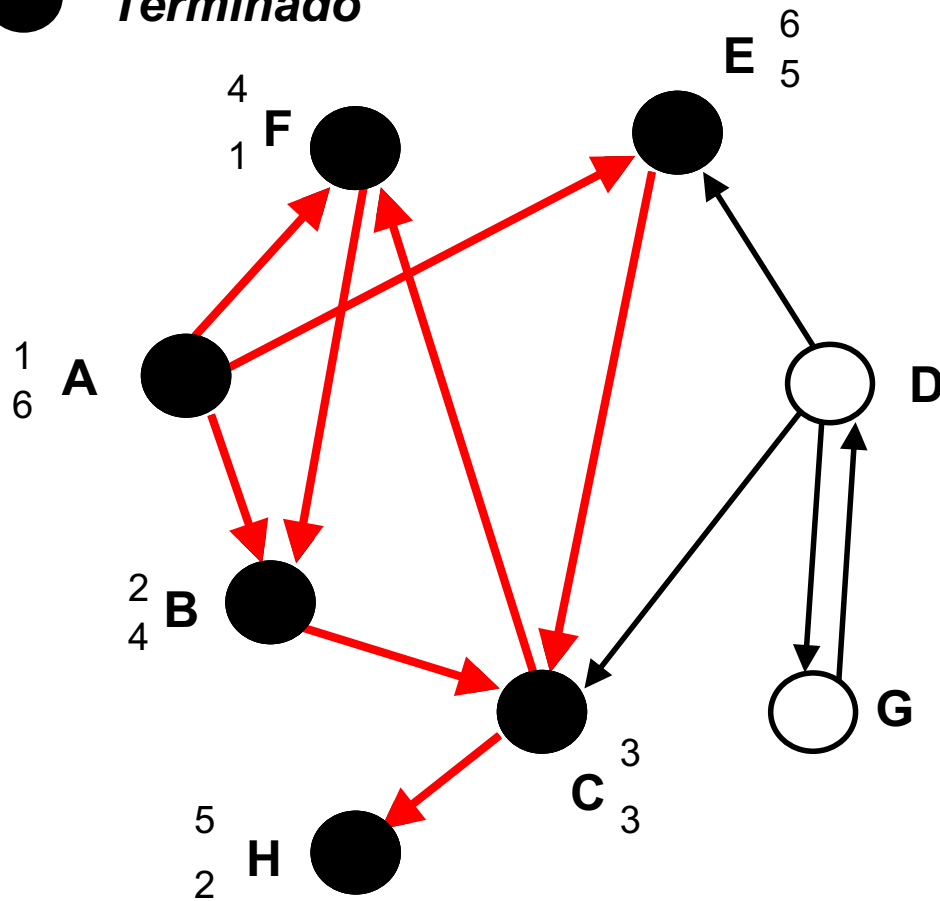
Exemplo

- Não visitado
- ◐ Em exploração
- Terminado



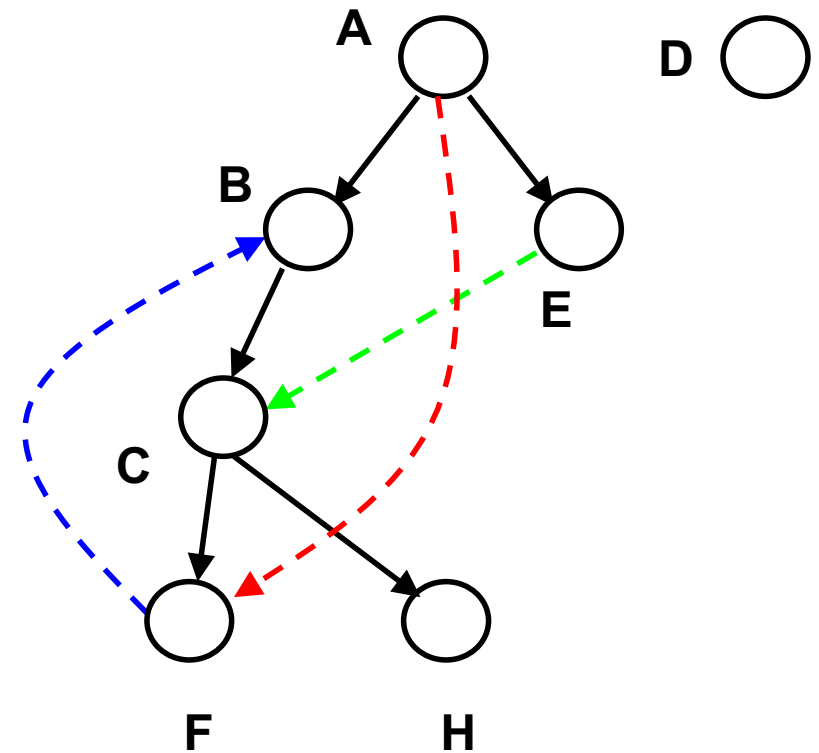
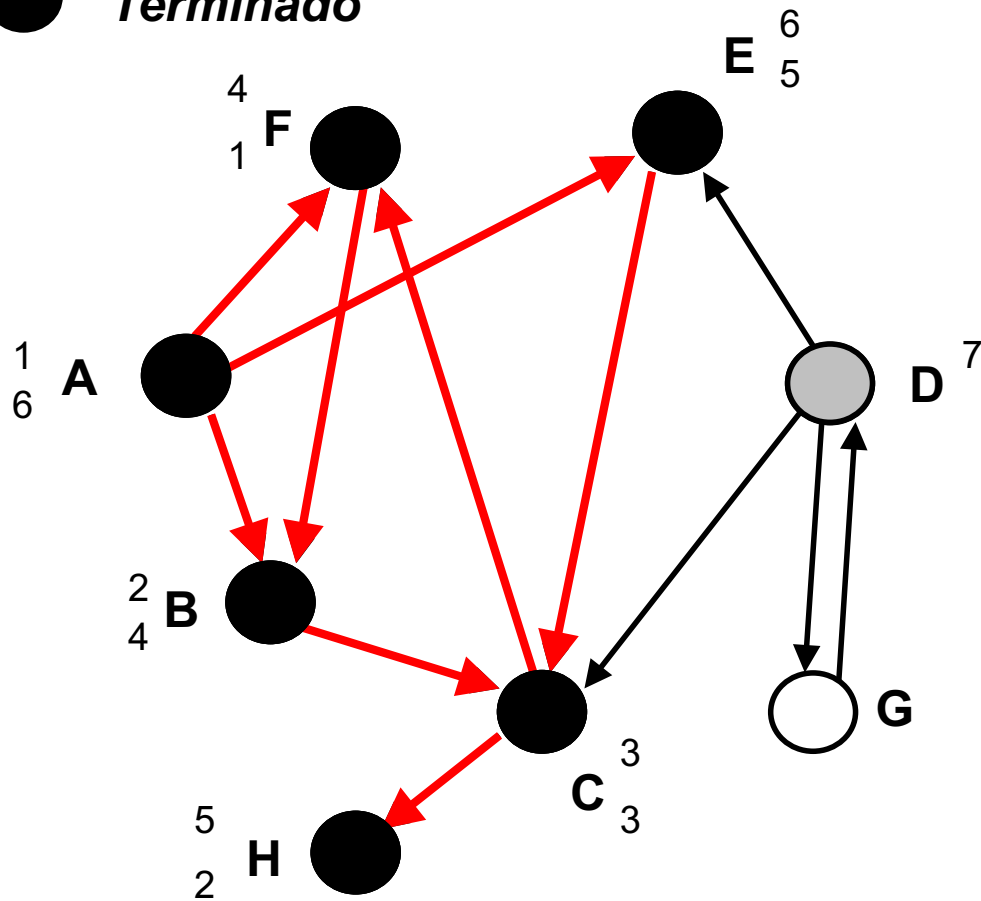
Exemplo

- Não visitado
- ◐ Em exploração
- Terminado



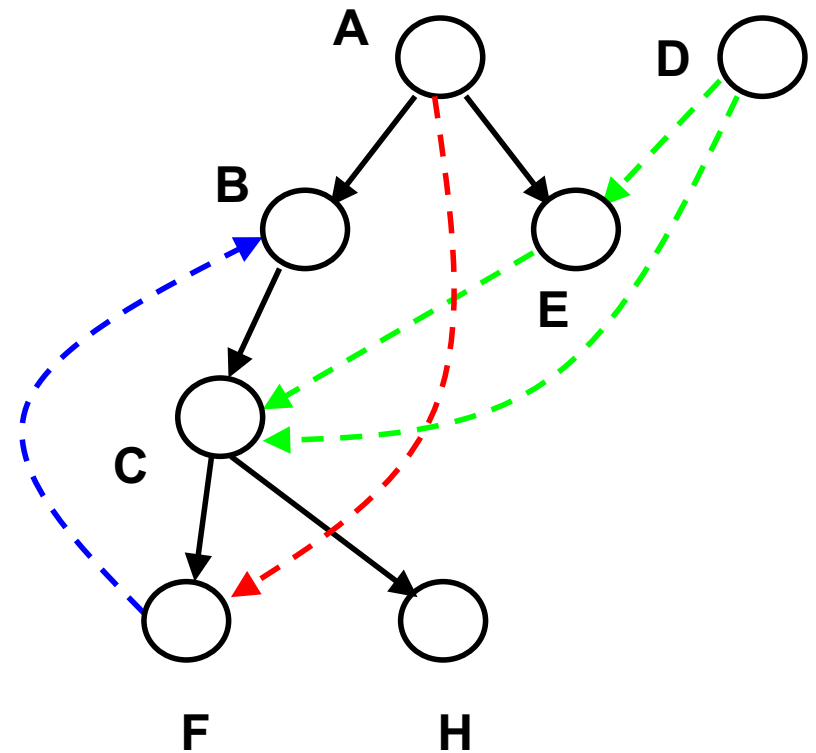
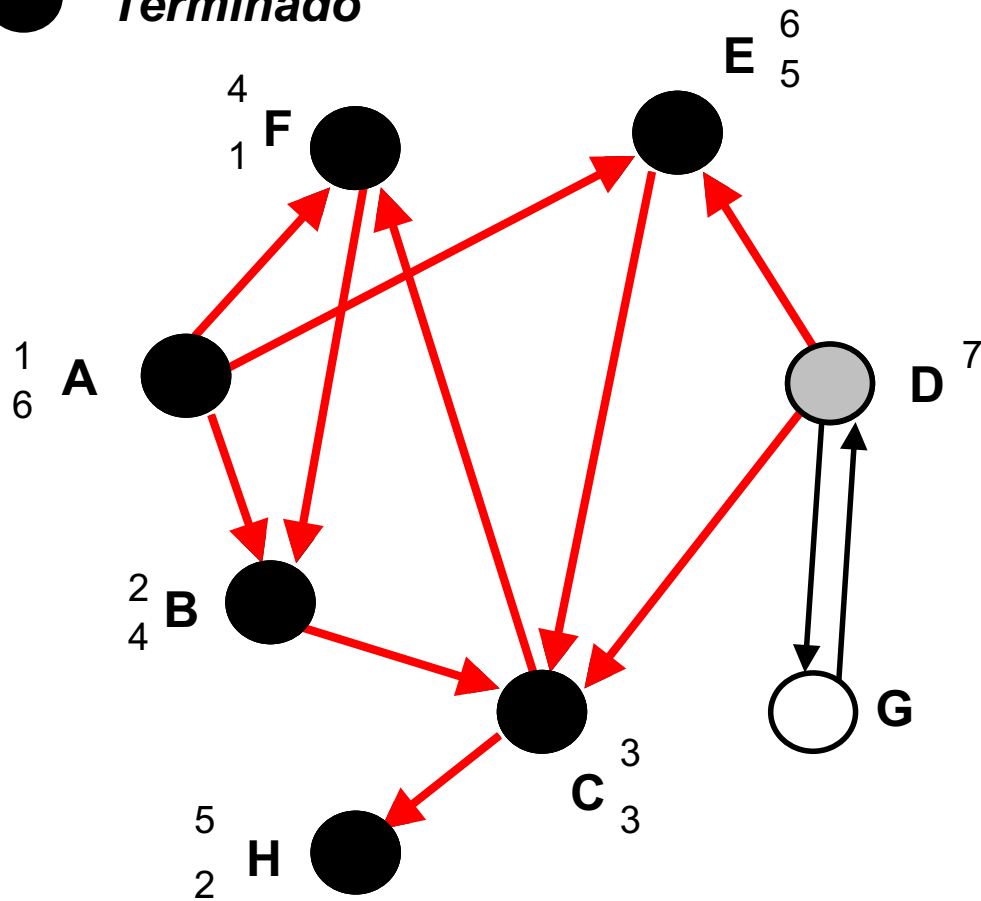
Exemplo

- Não visitado
- Em exploração
- Terminado



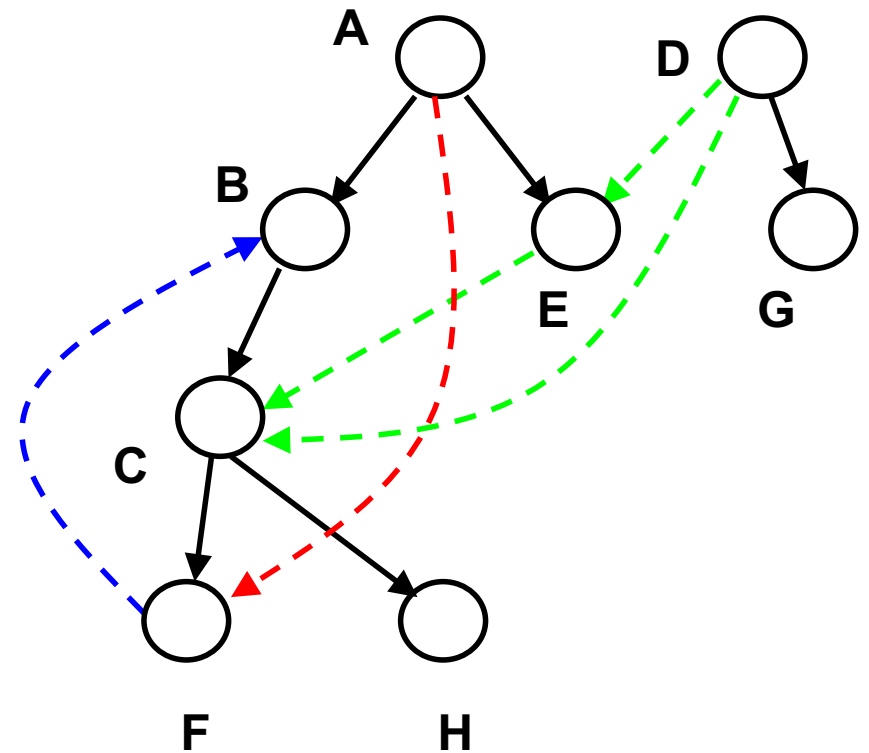
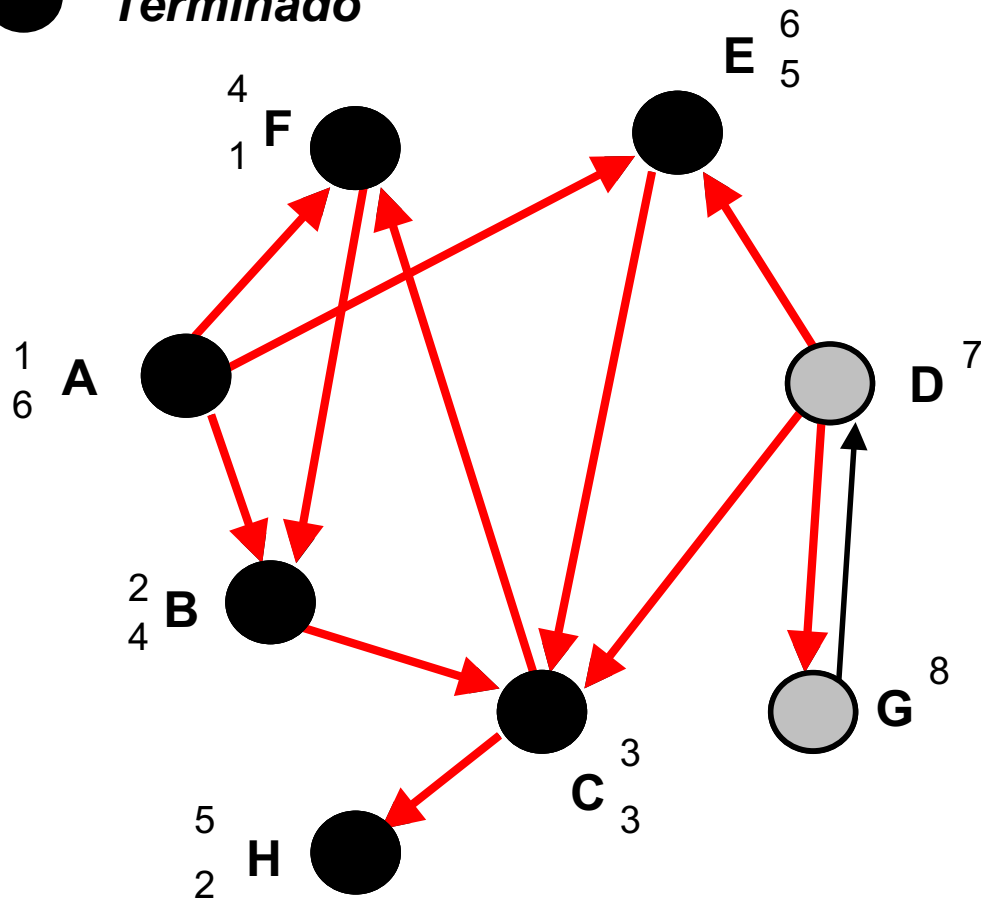
Exemplo

- Não visitado
- Em exploração
- Terminado



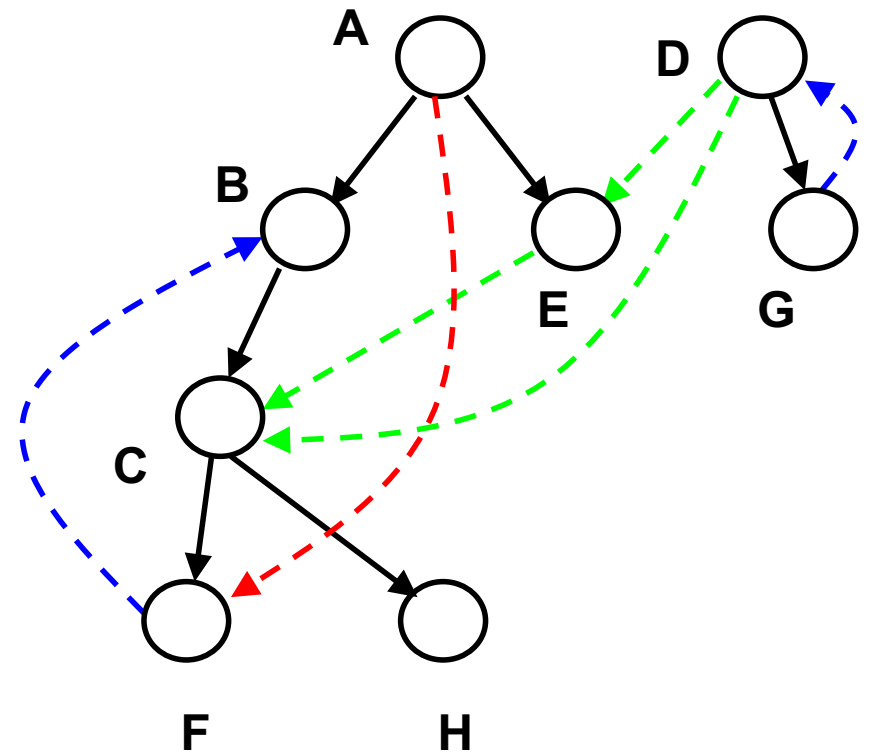
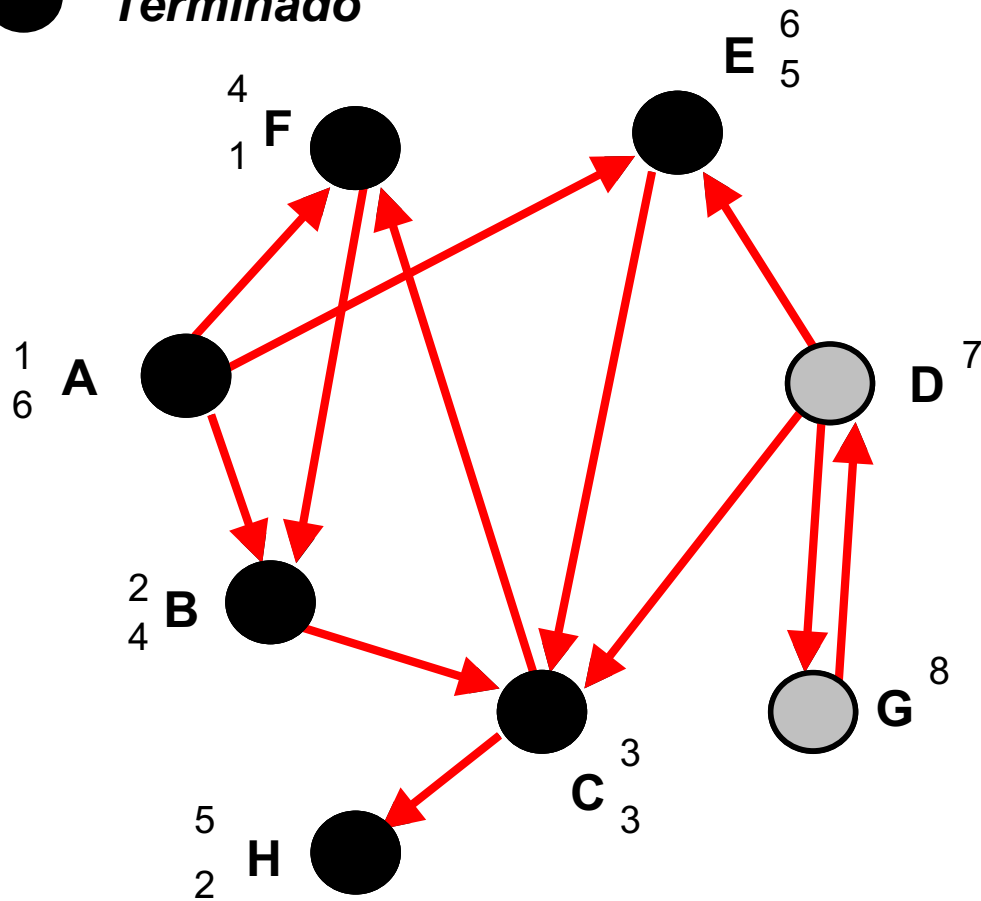
Exemplo

- *Não visitado*
- ◐ *Em exploração*
- *Terminado*



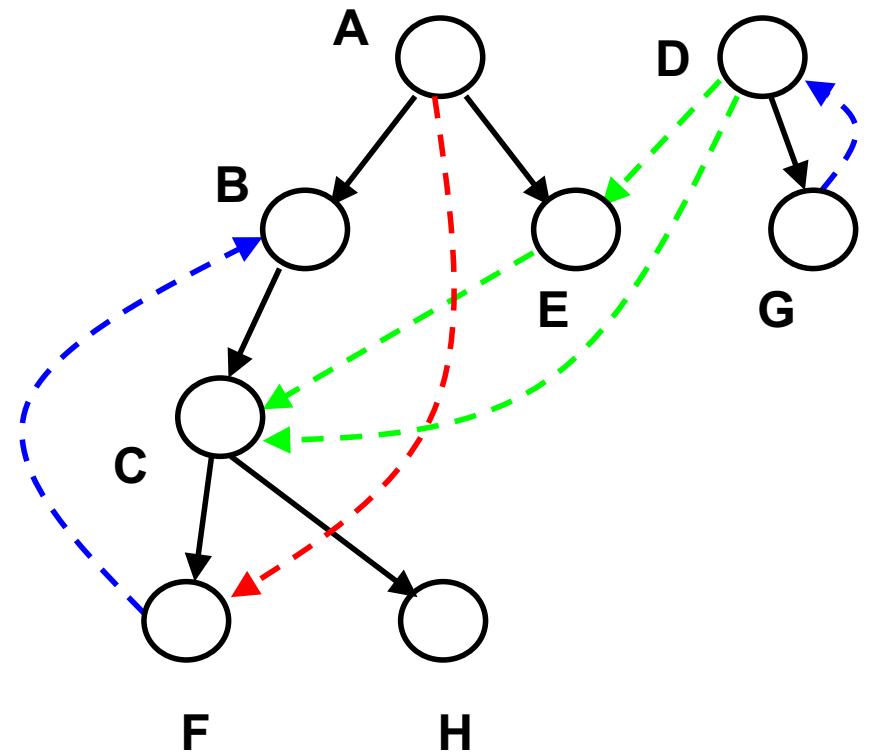
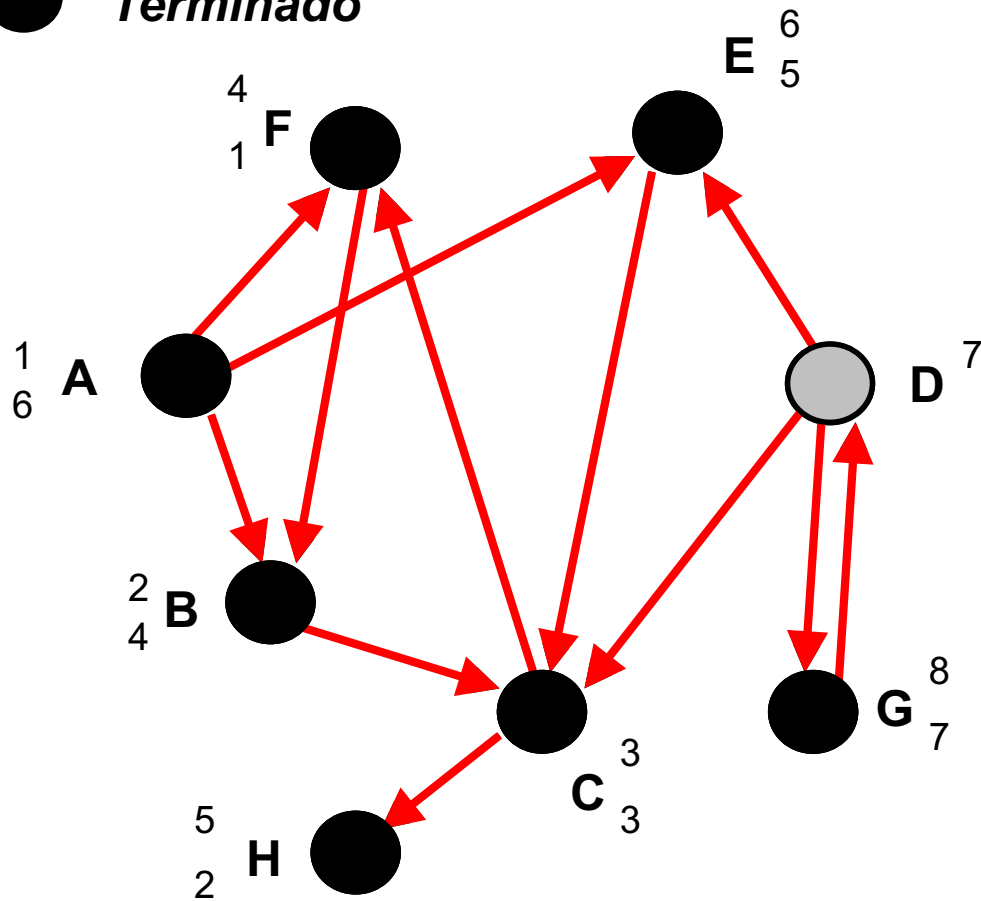
Exemplo

- *Não visitado*
- ◐ *Em exploração*
- *Terminado*



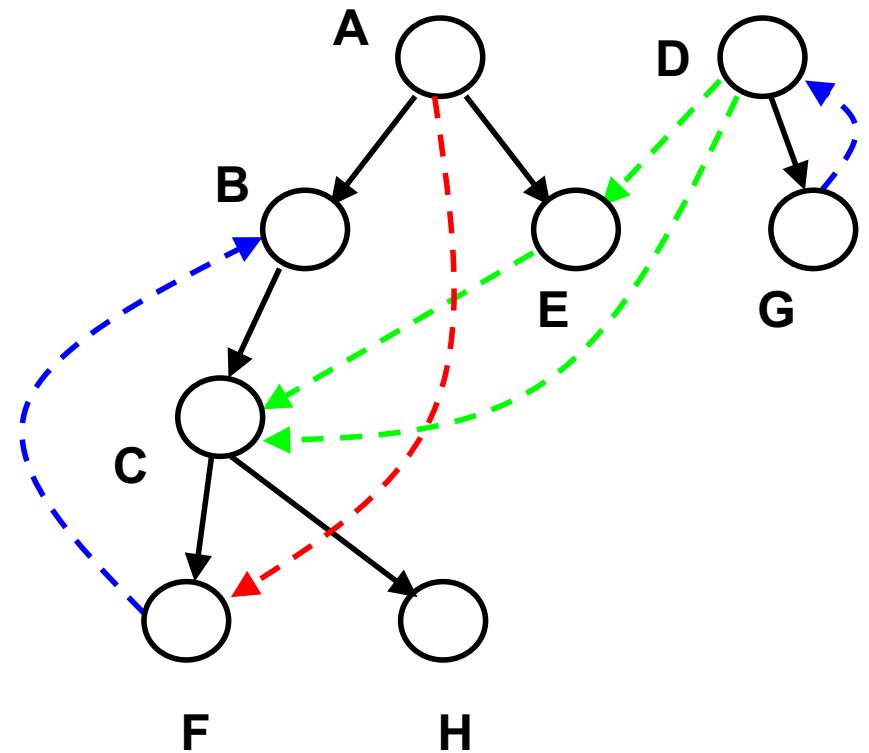
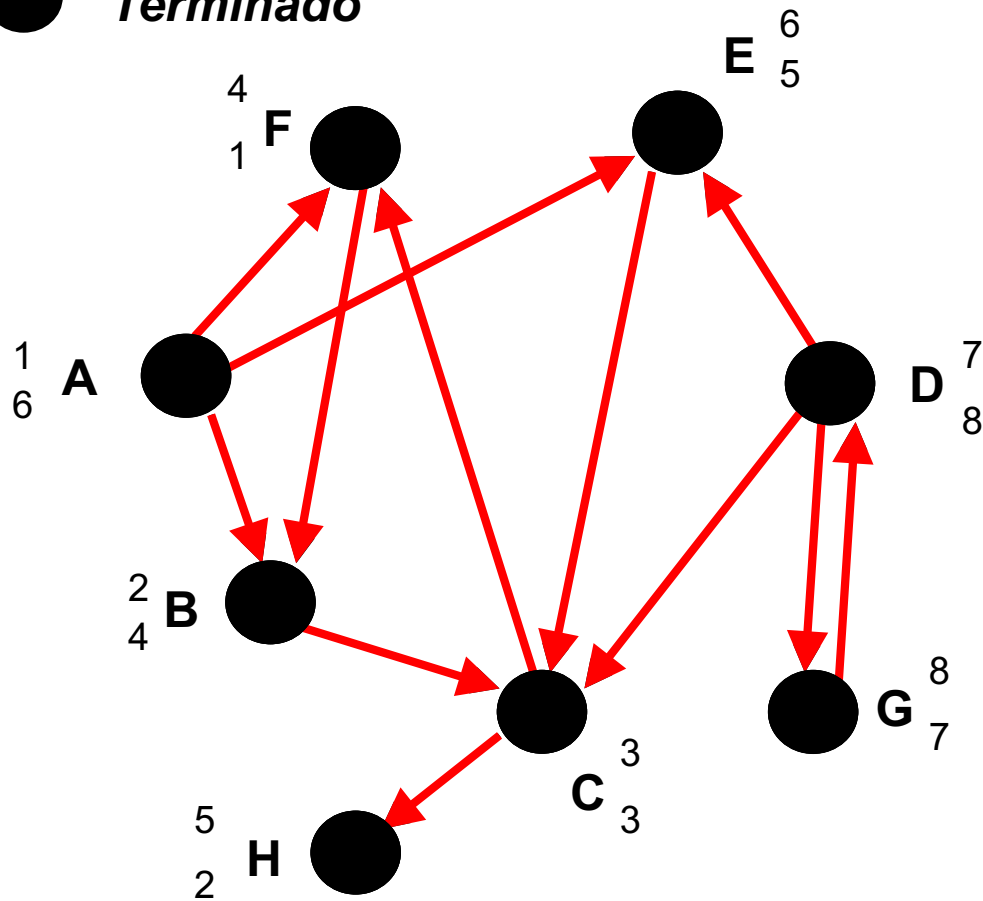
Exemplo

- *Não visitado*
- ◐ *Em exploração*
- *Terminado*



Exemplo

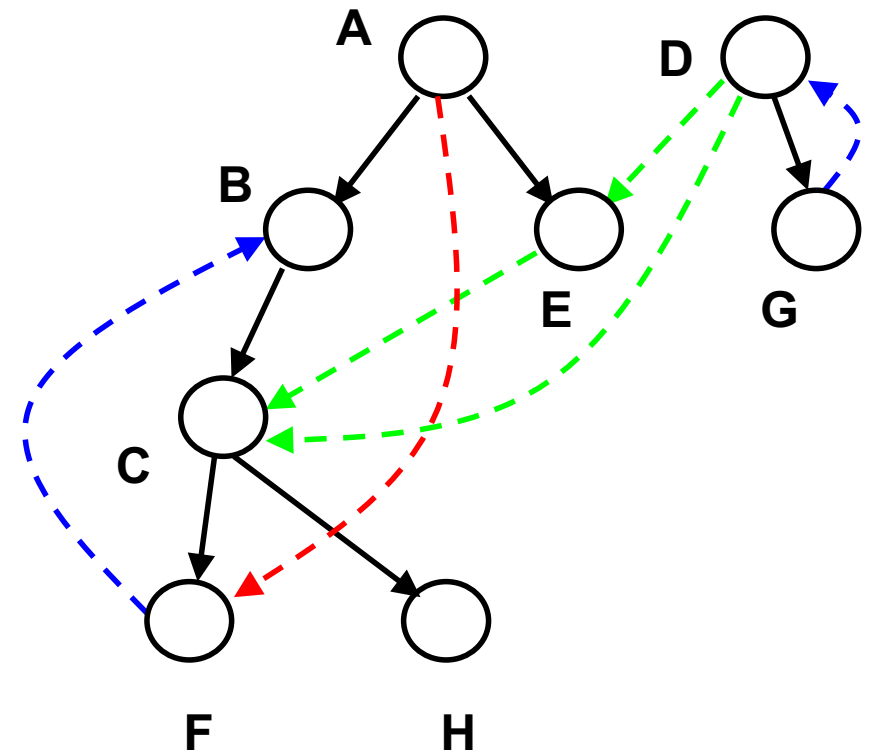
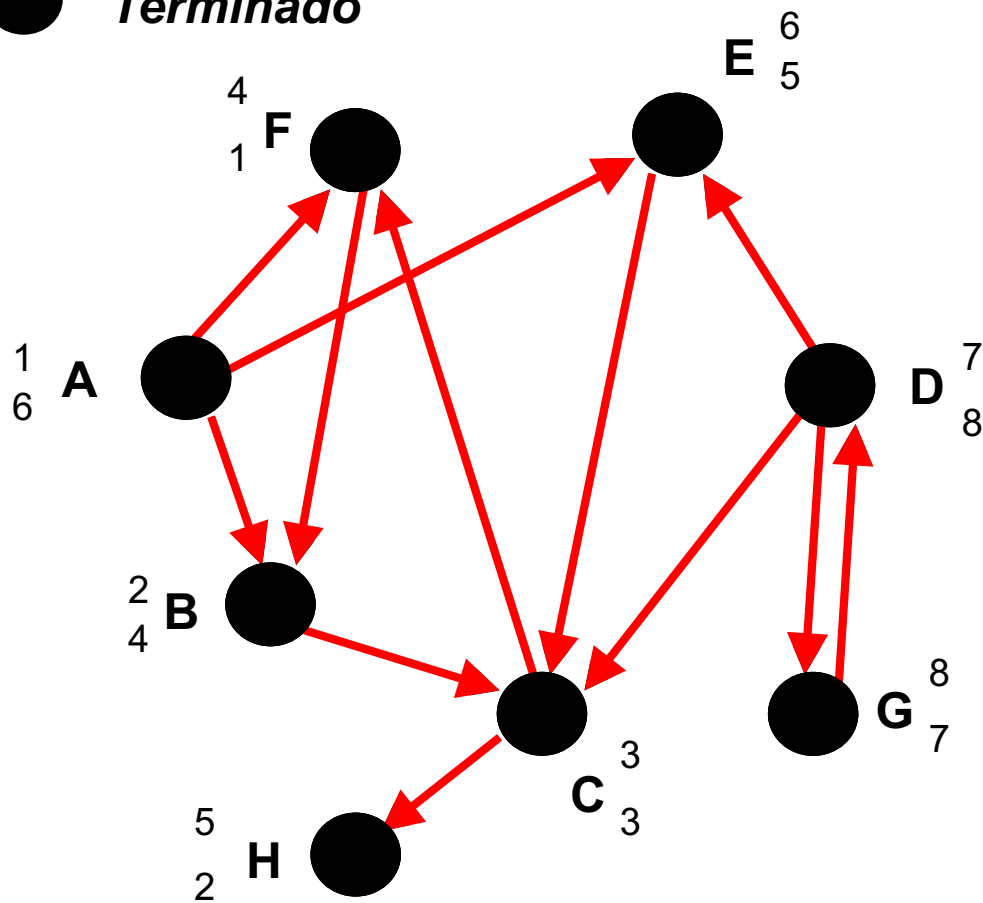
- *Não visitado*
- ◐ *Em exploração*
- *Terminado*



Exemplo

- Não visitado
- ◐ Em exploração
- Terminado

exp1: número de exploração

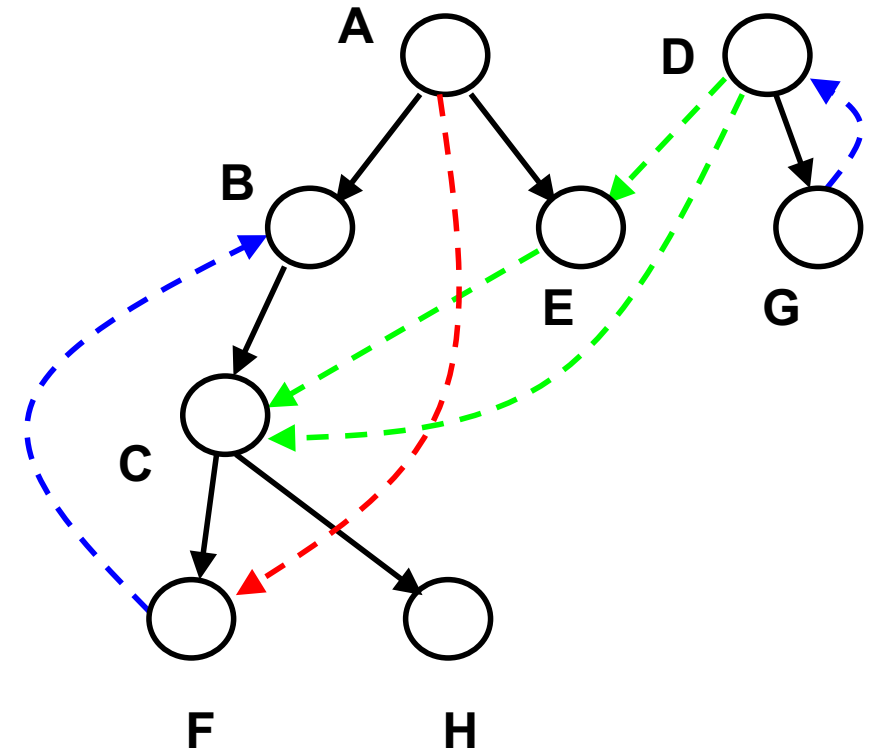
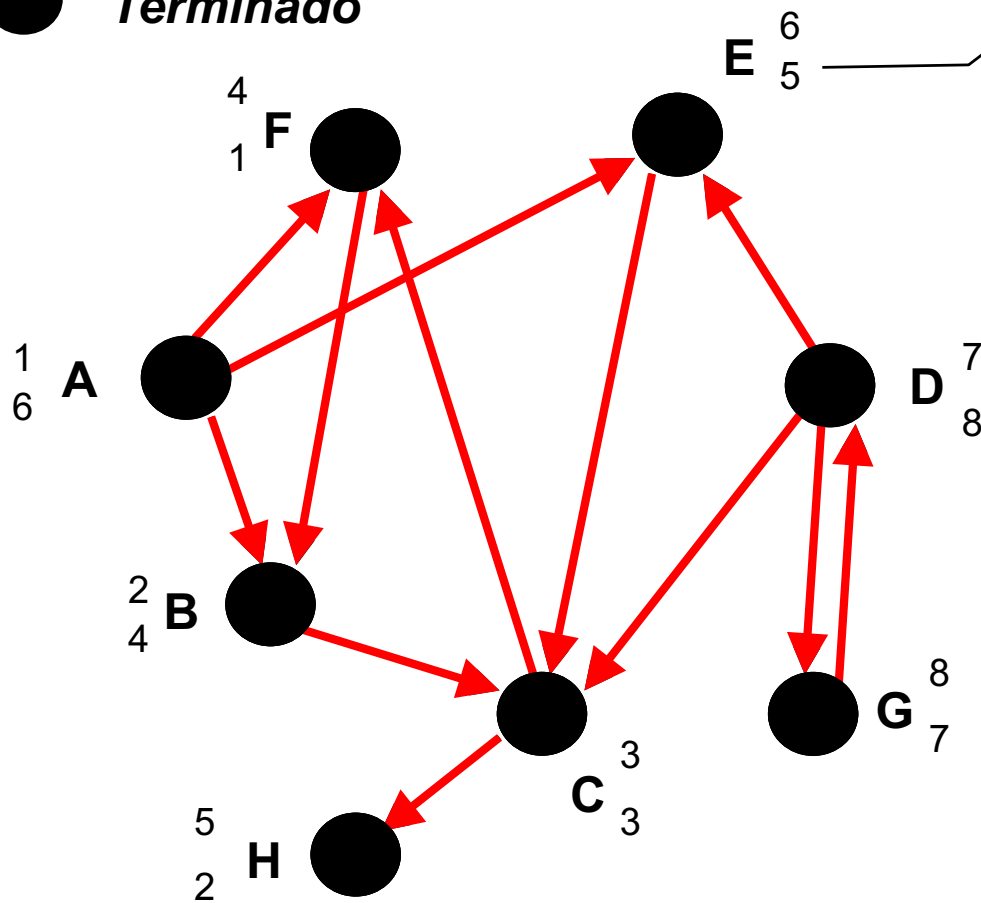


Exemplo

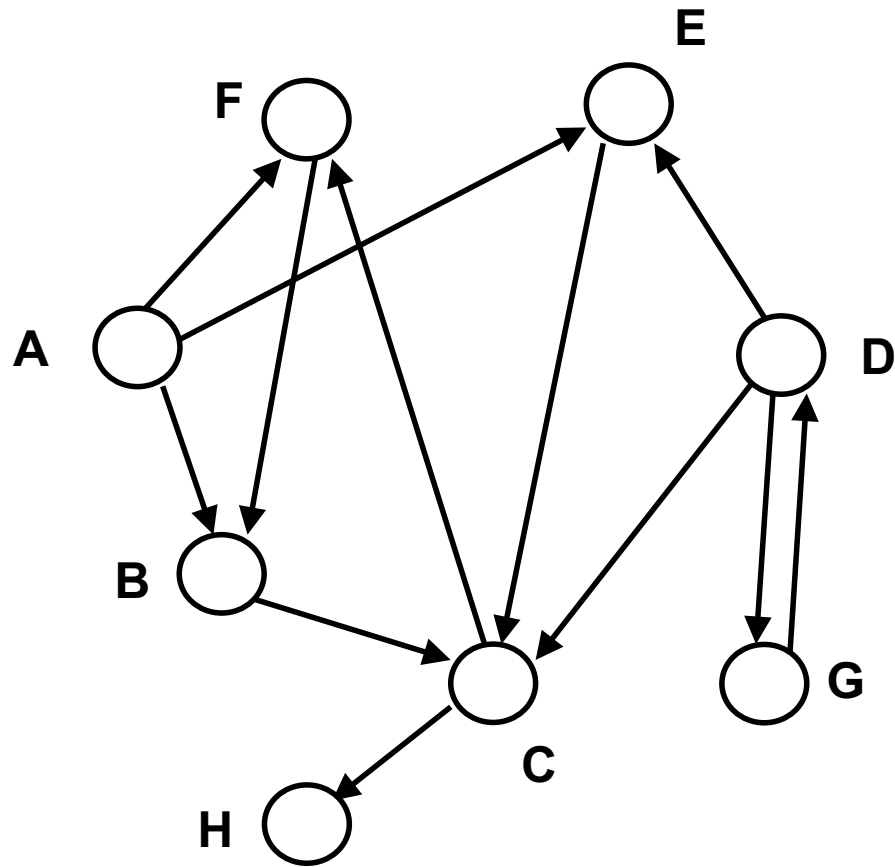
- Não visitado
- ◐ Em exploração
- Terminado

exp1: número de exploração

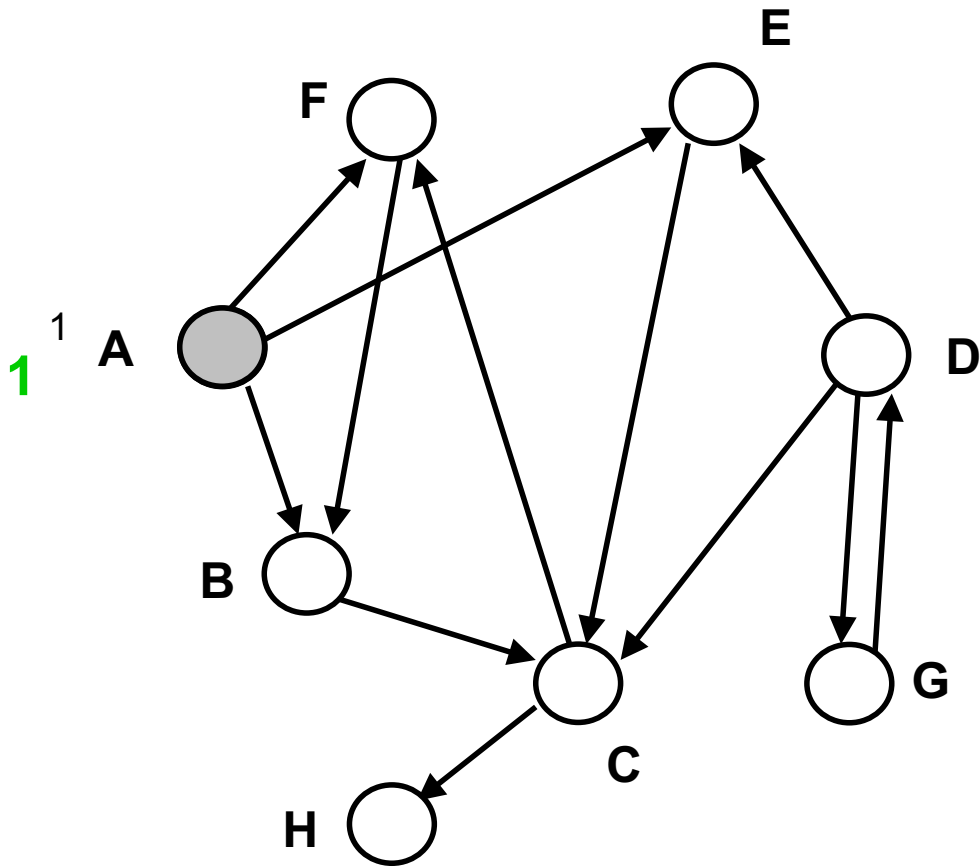
comp: número de complementação



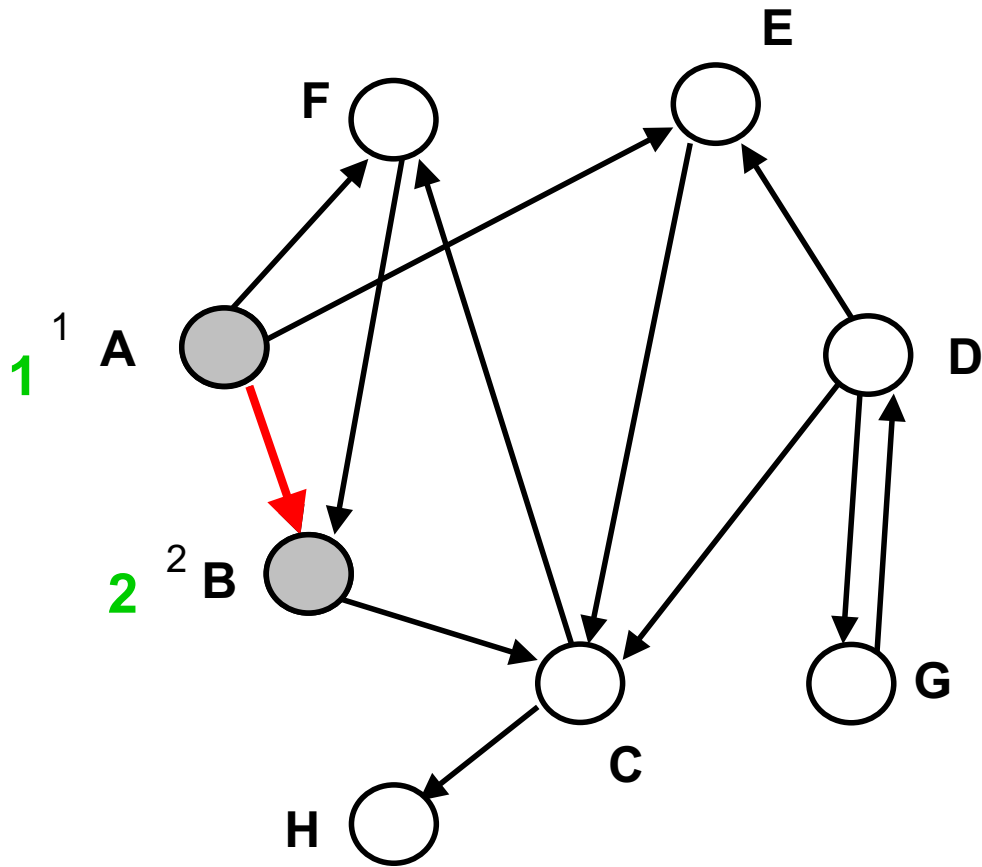
Exemplo



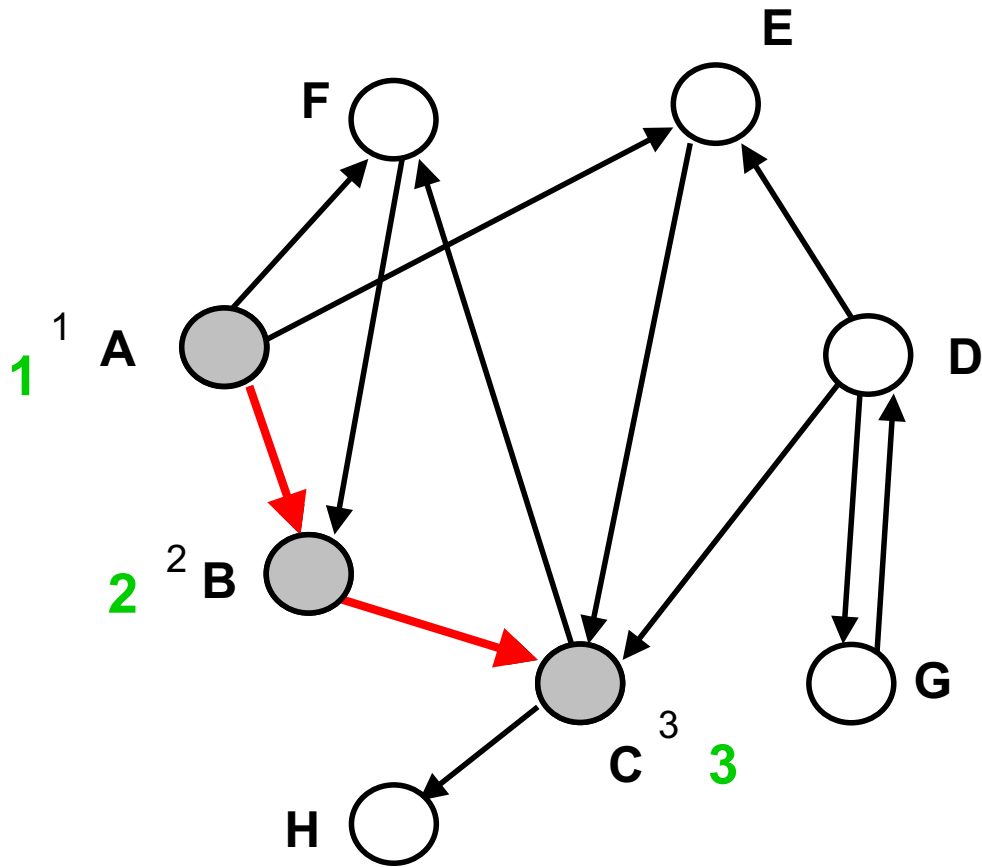
Exemplo



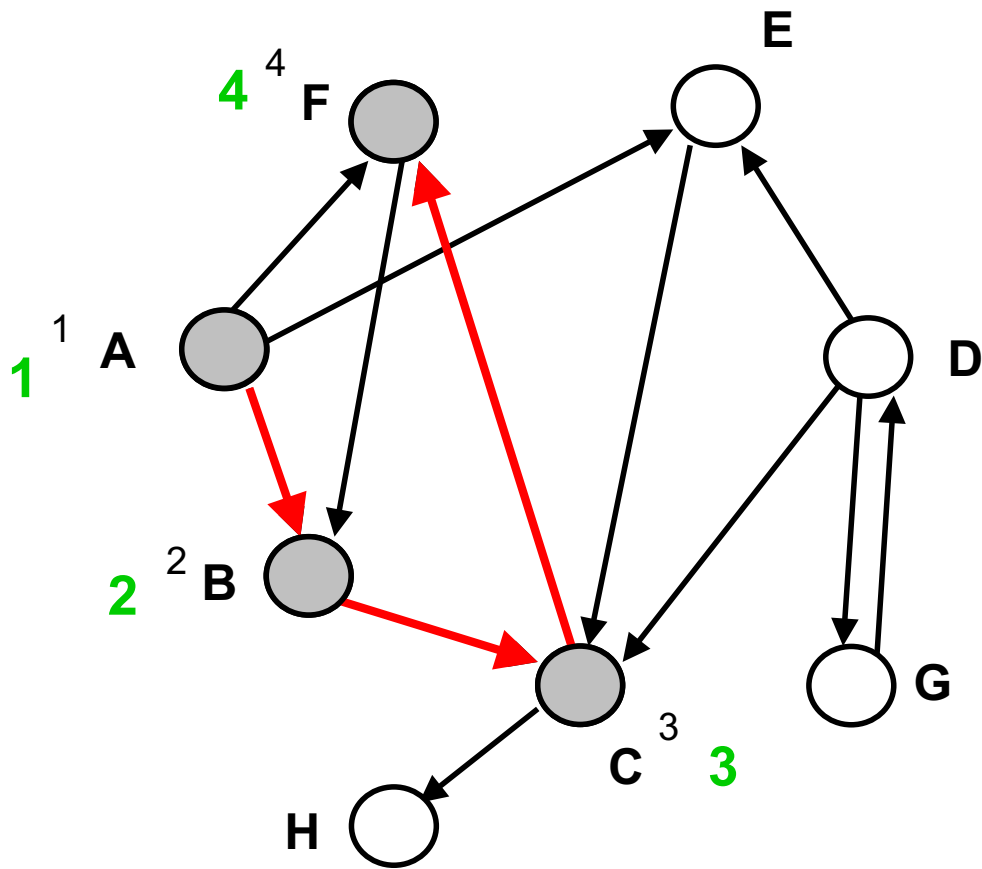
Exemplo



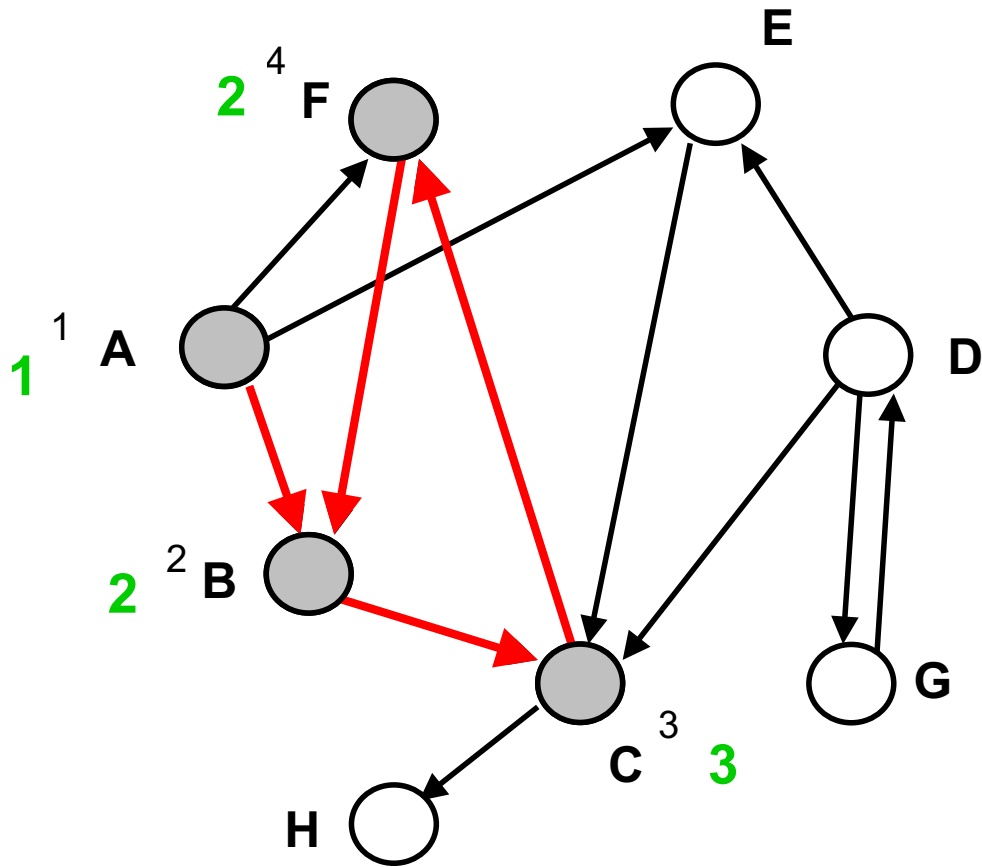
Exemplo



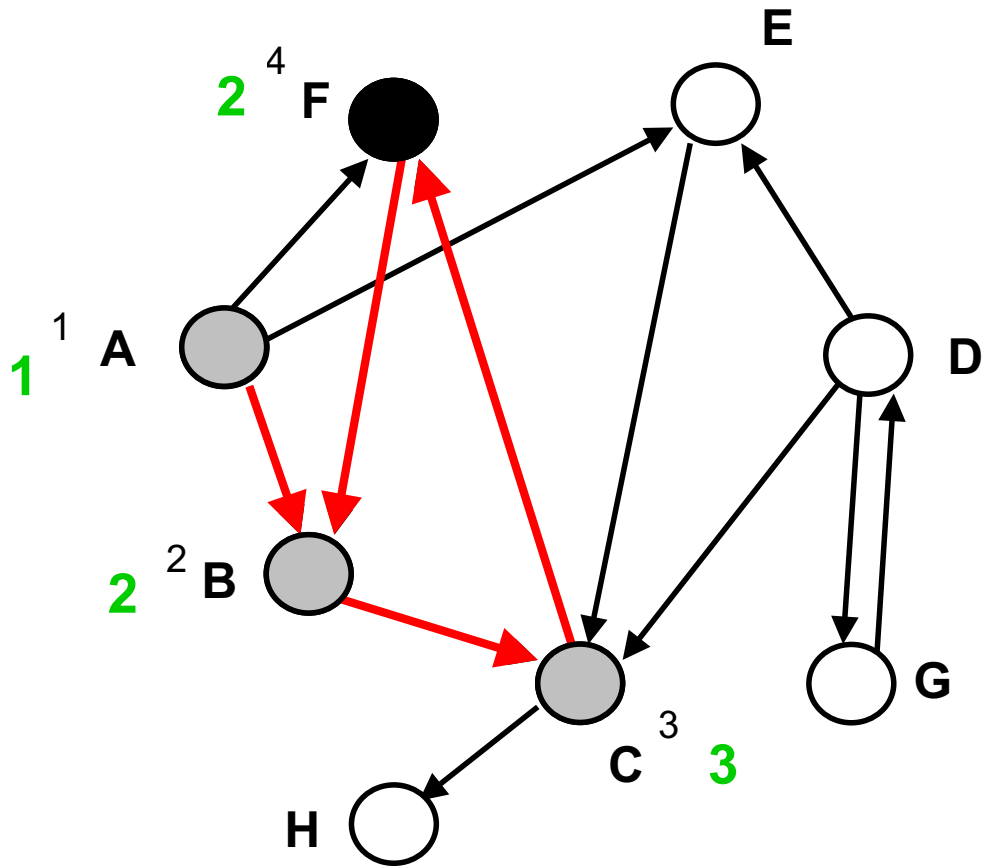
Exemplo



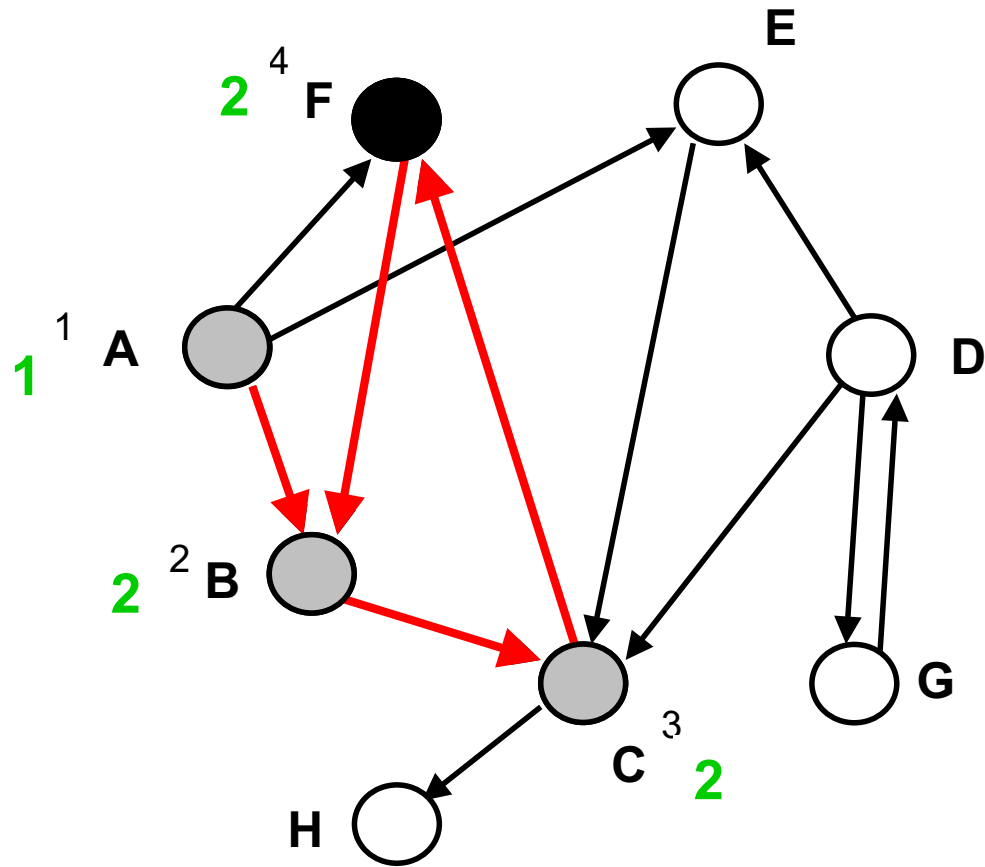
Exemplo



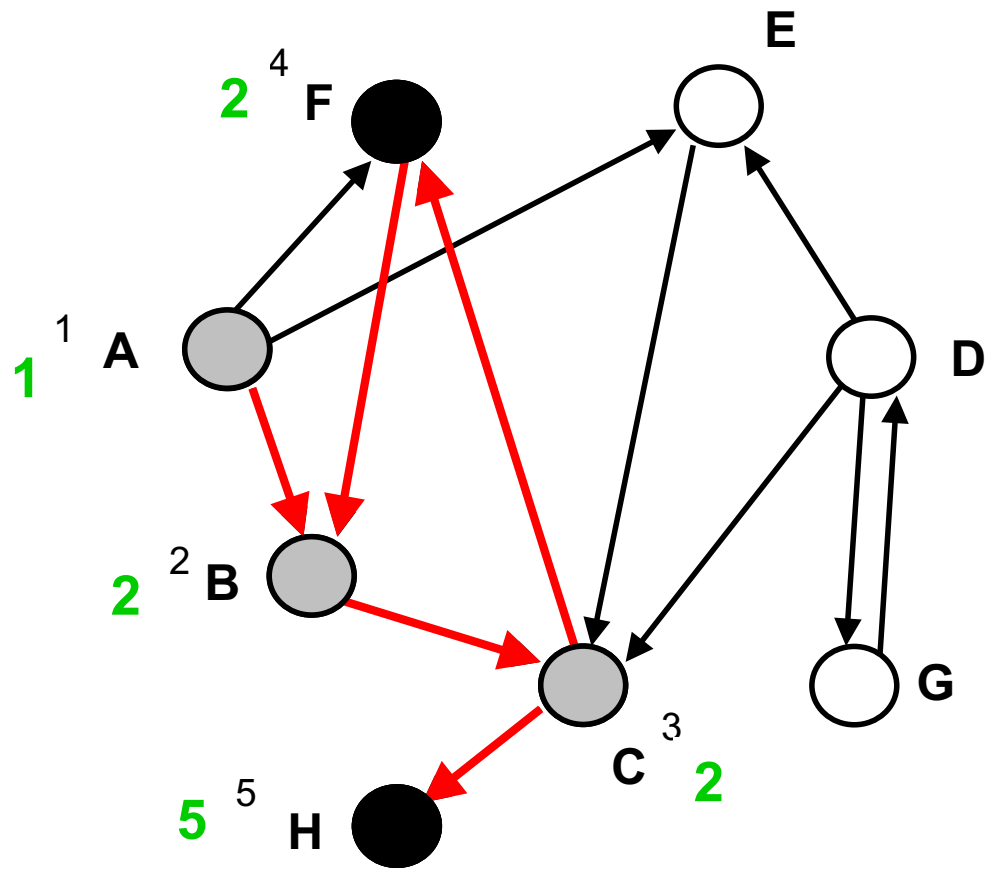
Exemplo



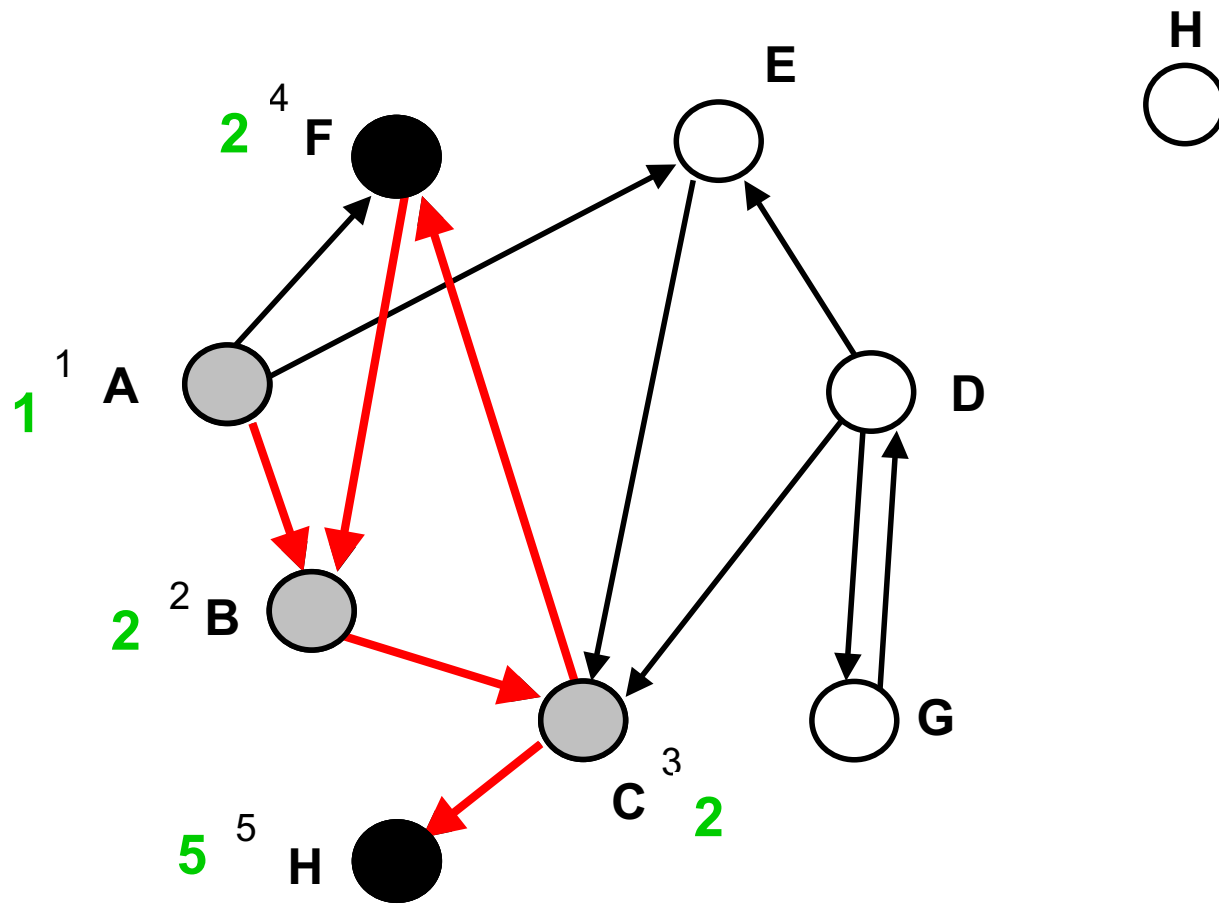
Exemplo



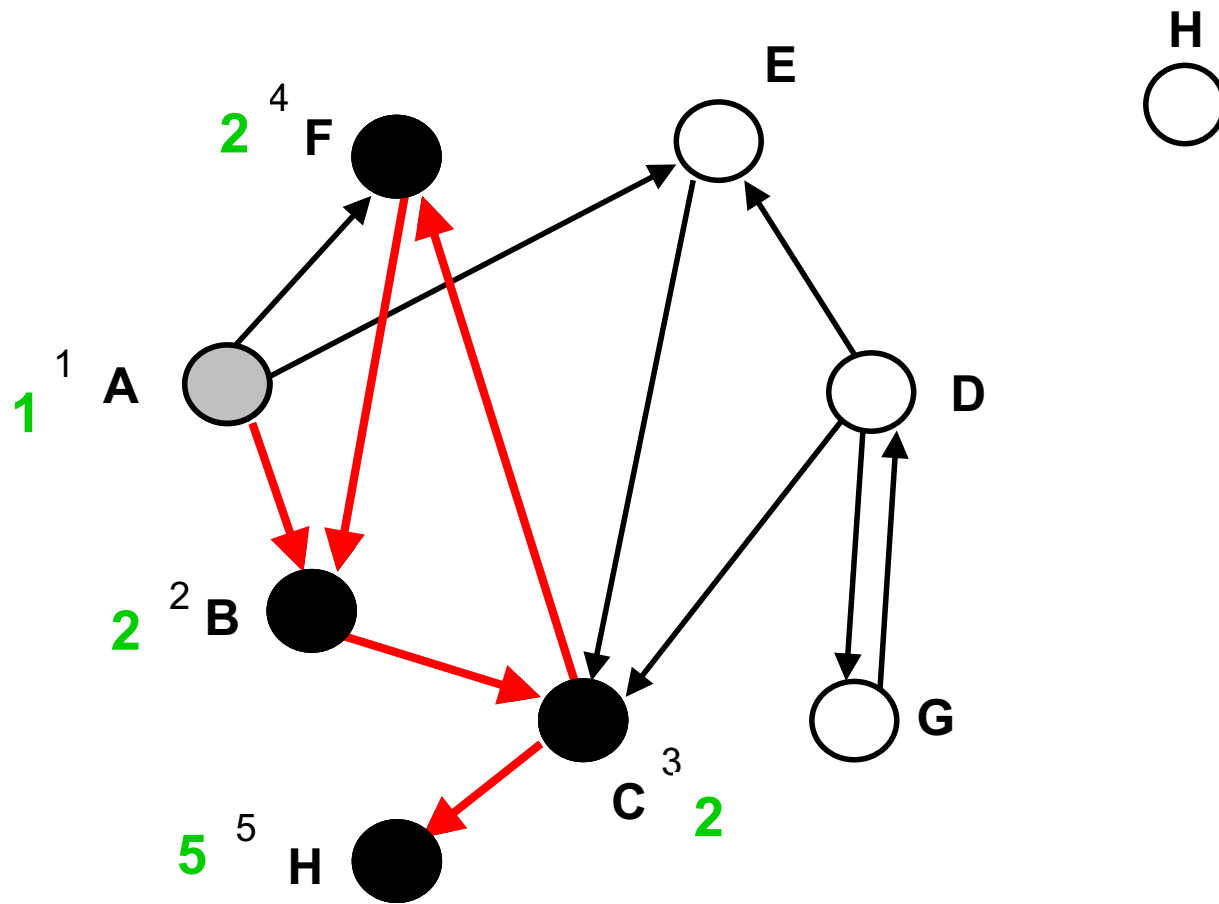
Exemplo



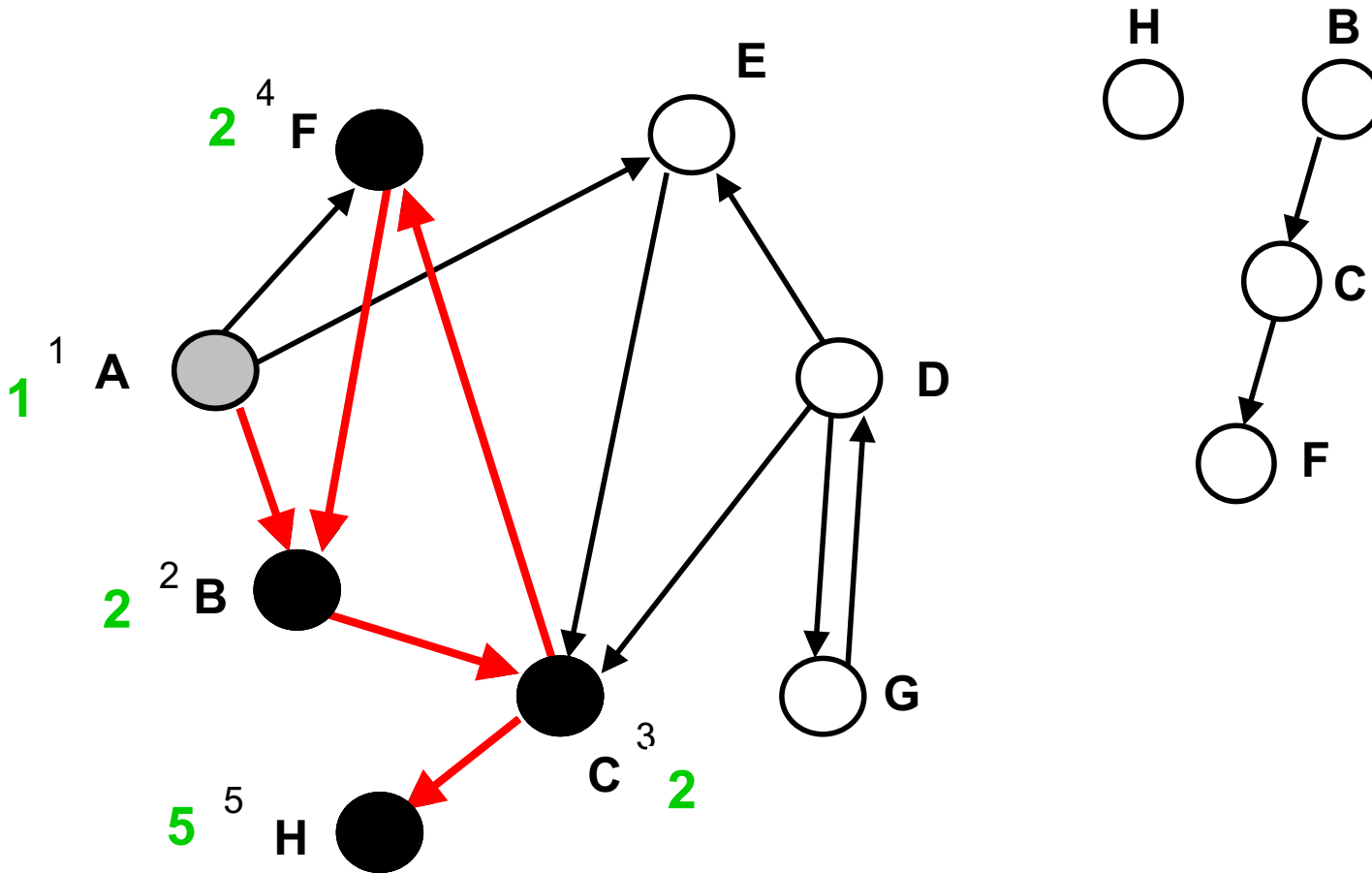
Exemplo



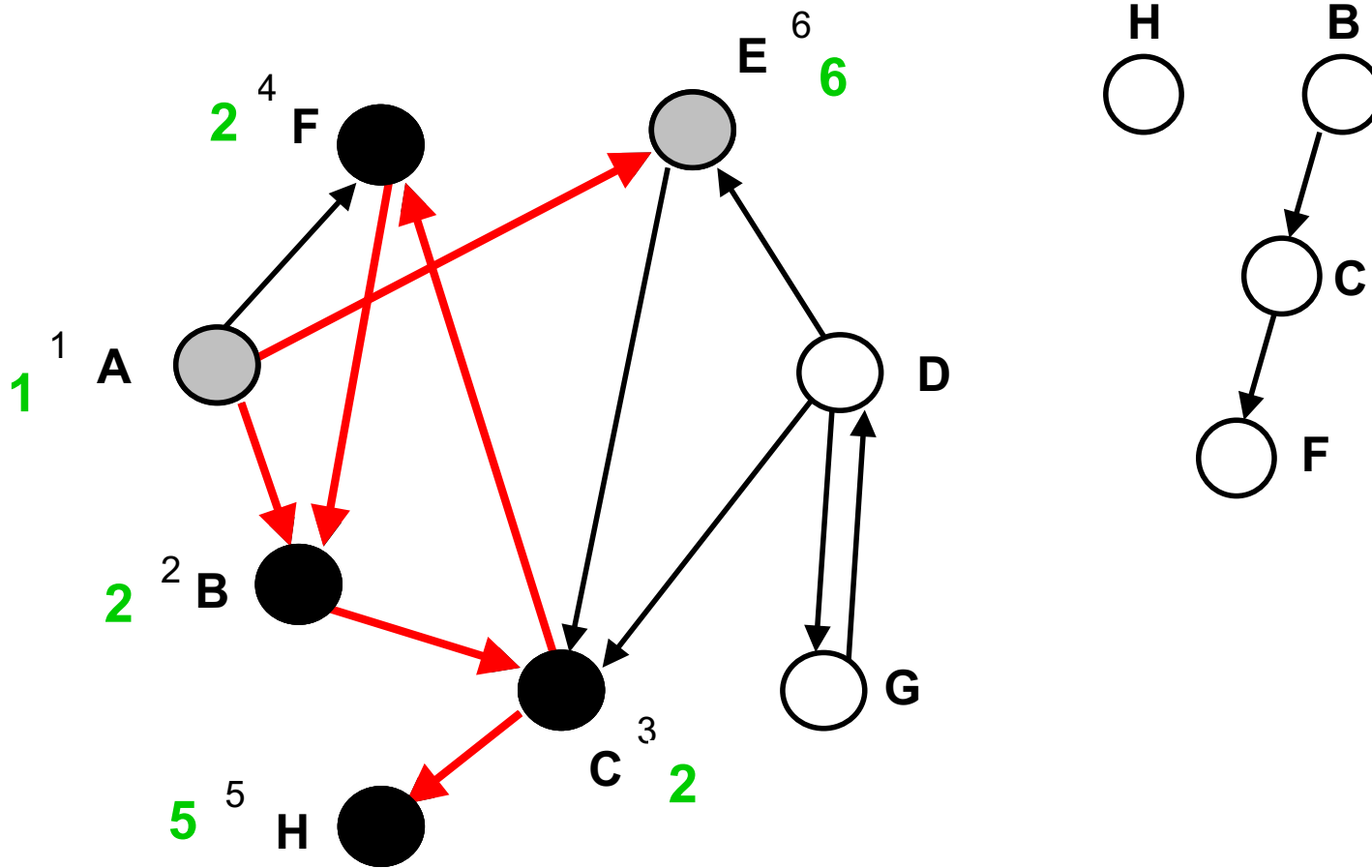
Exemplo



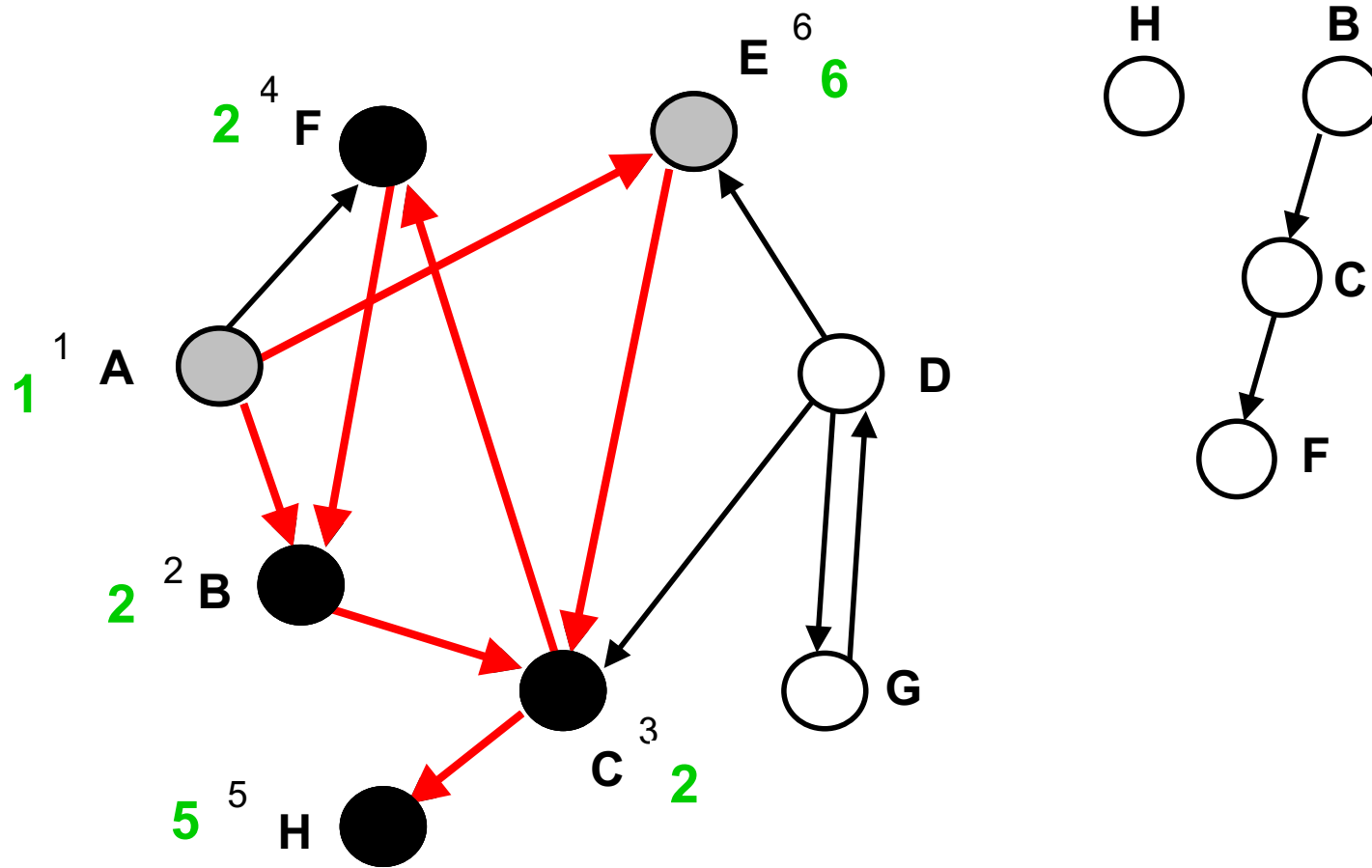
Exemplo



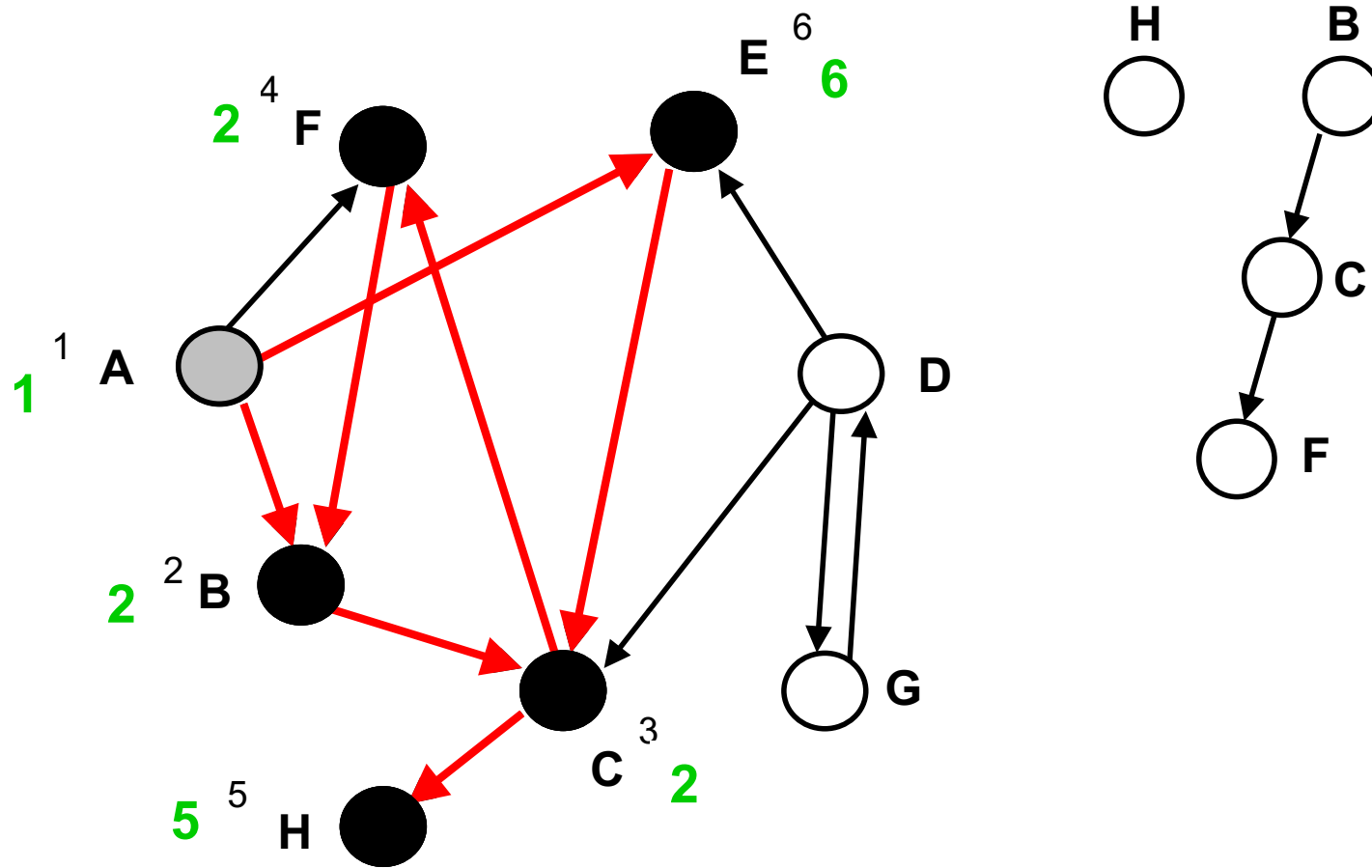
Exemplo



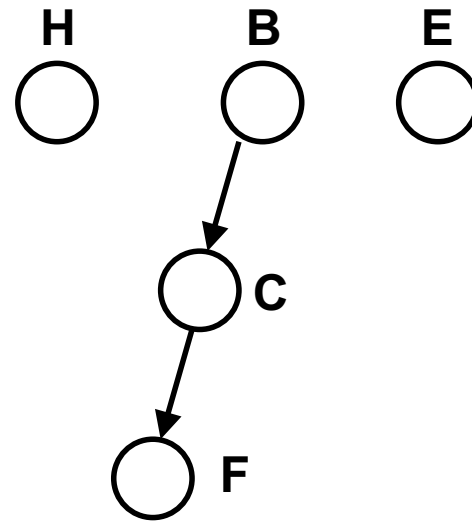
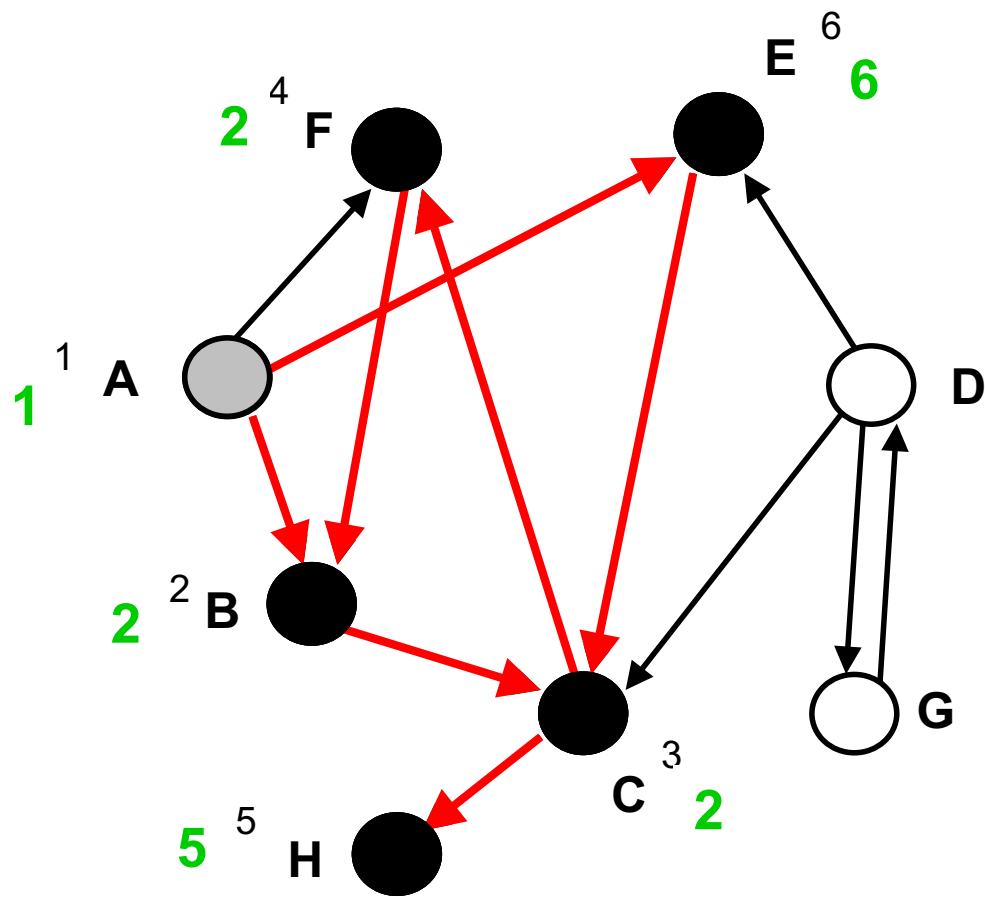
Exemplo



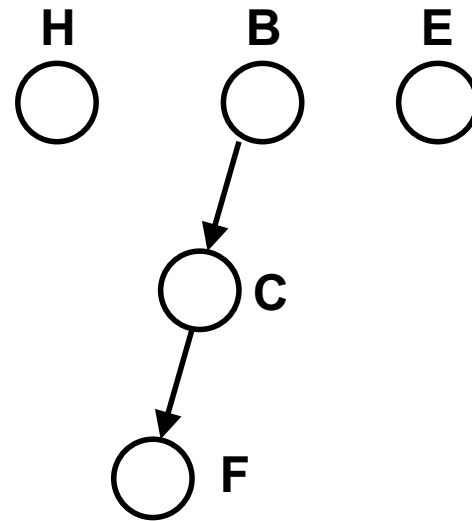
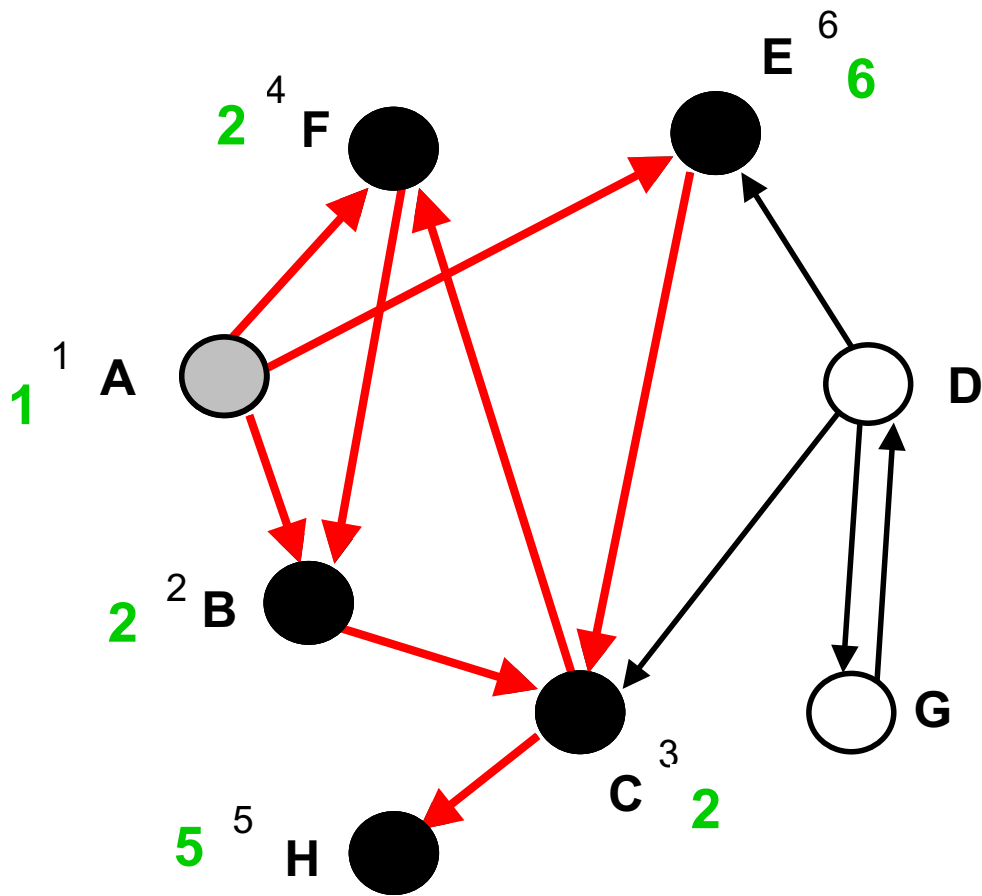
Exemplo



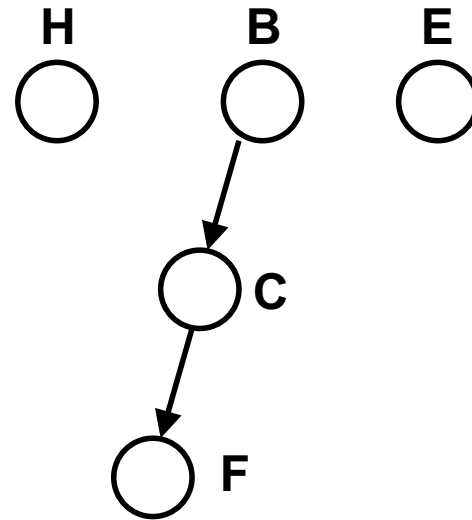
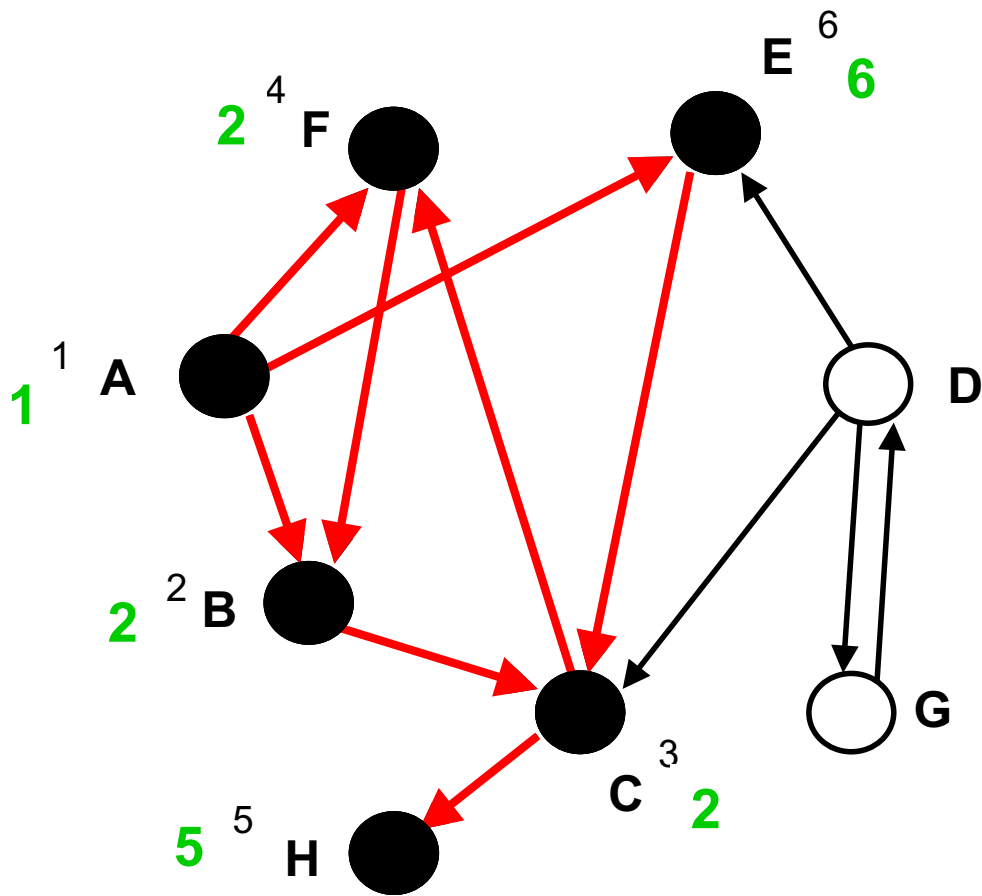
Exemplo



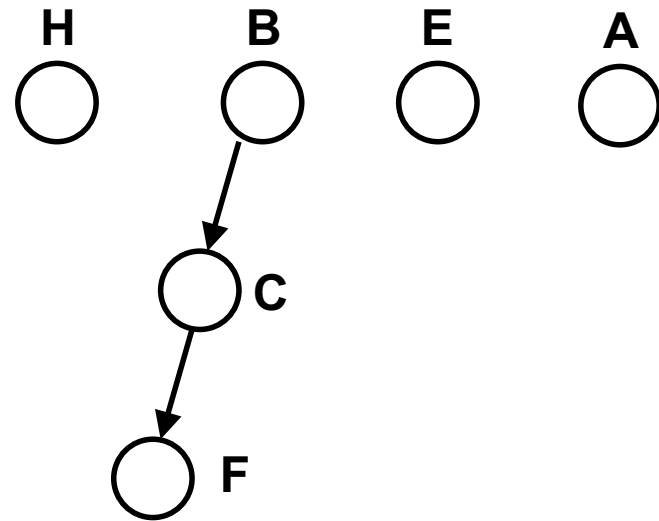
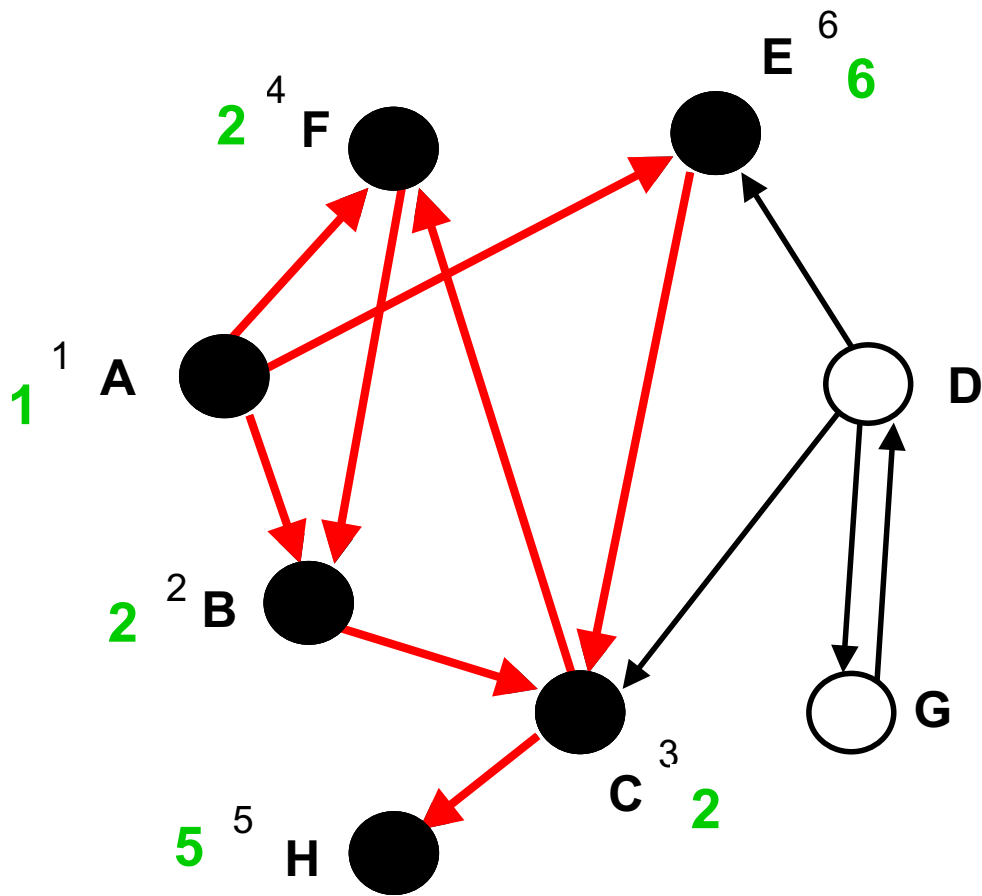
Exemplo



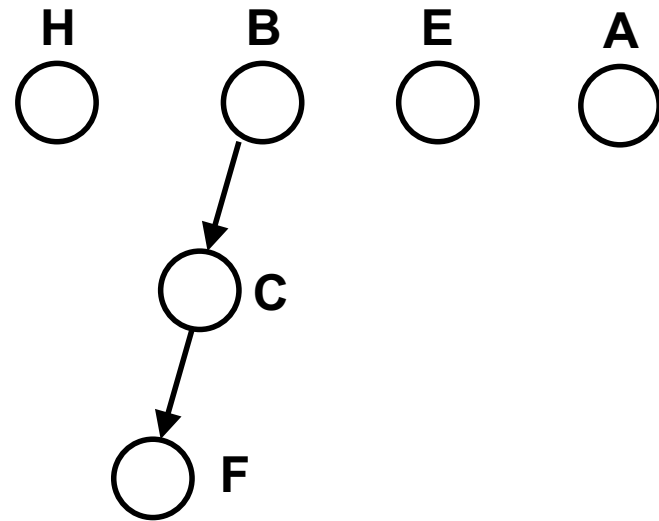
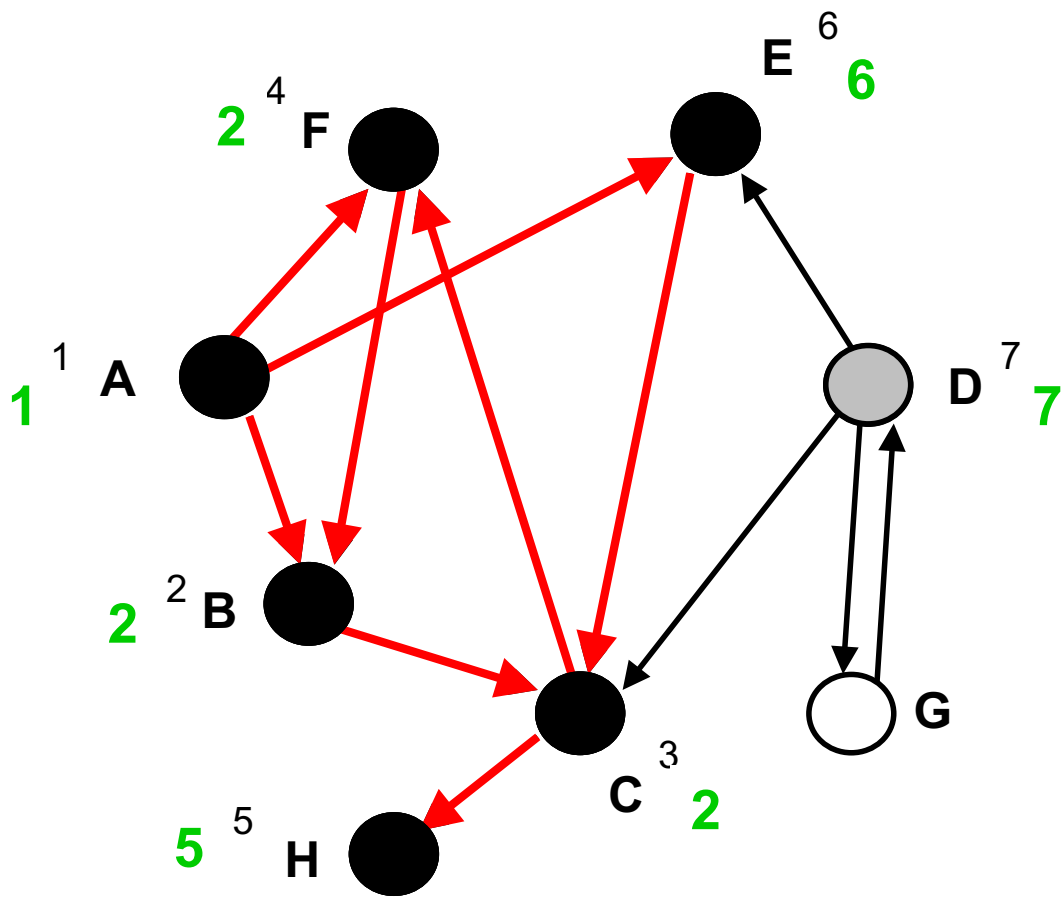
Exemplo



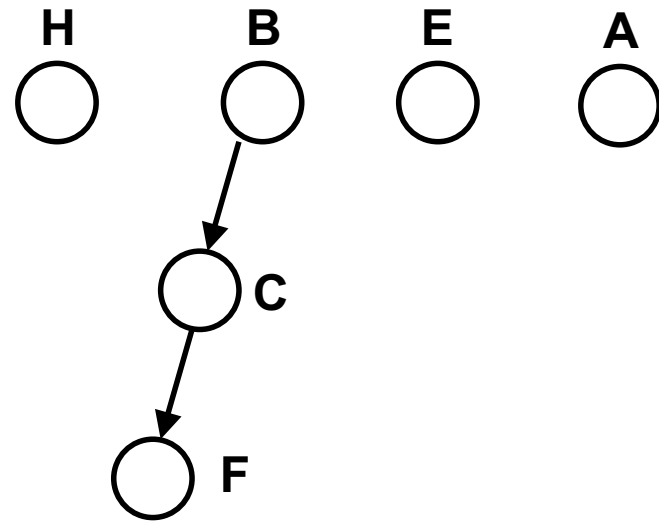
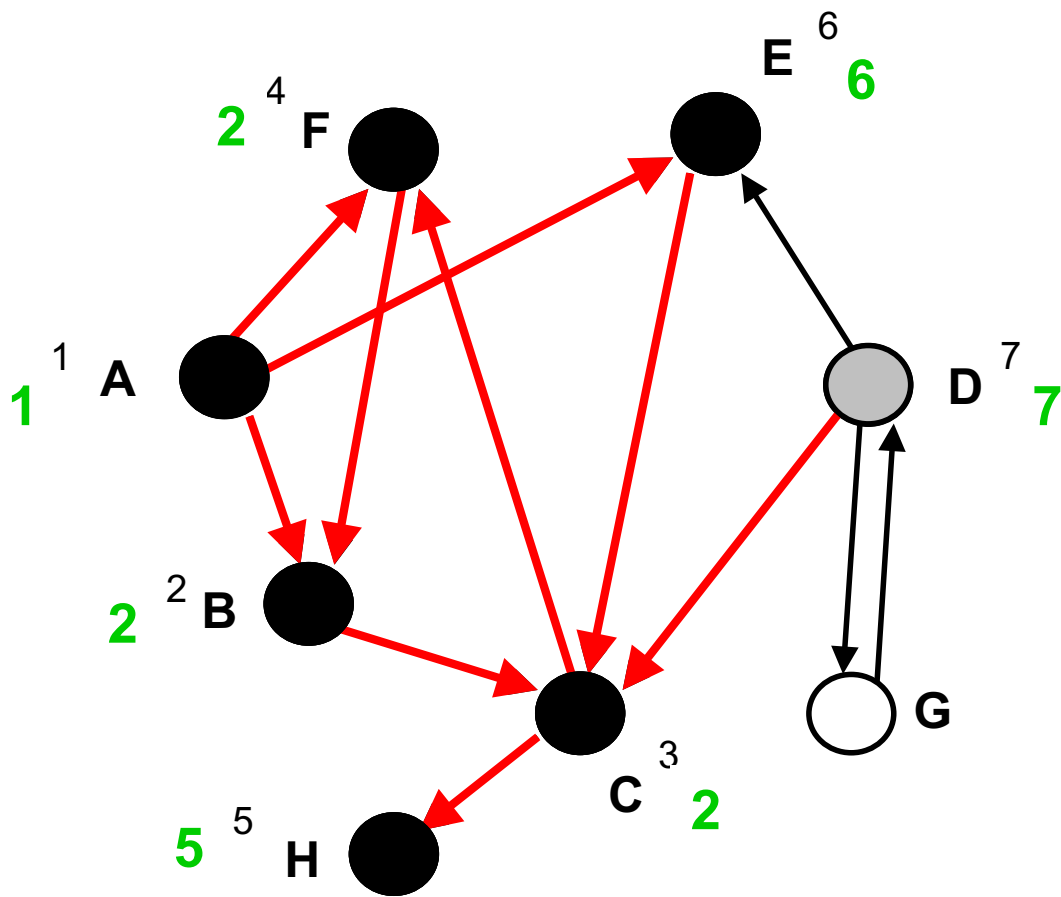
Exemplo



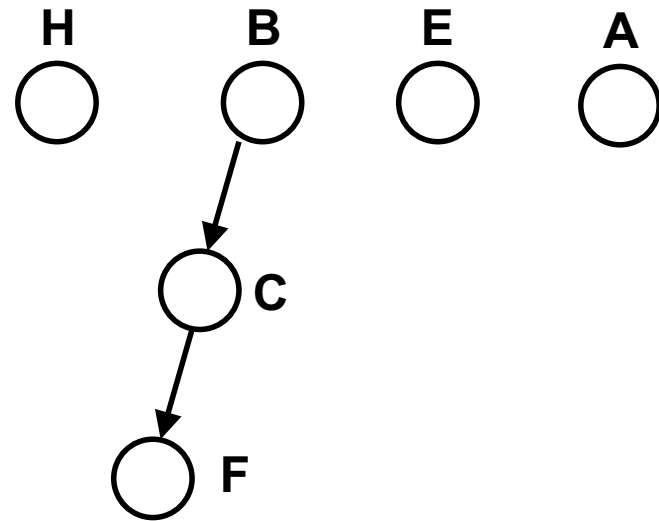
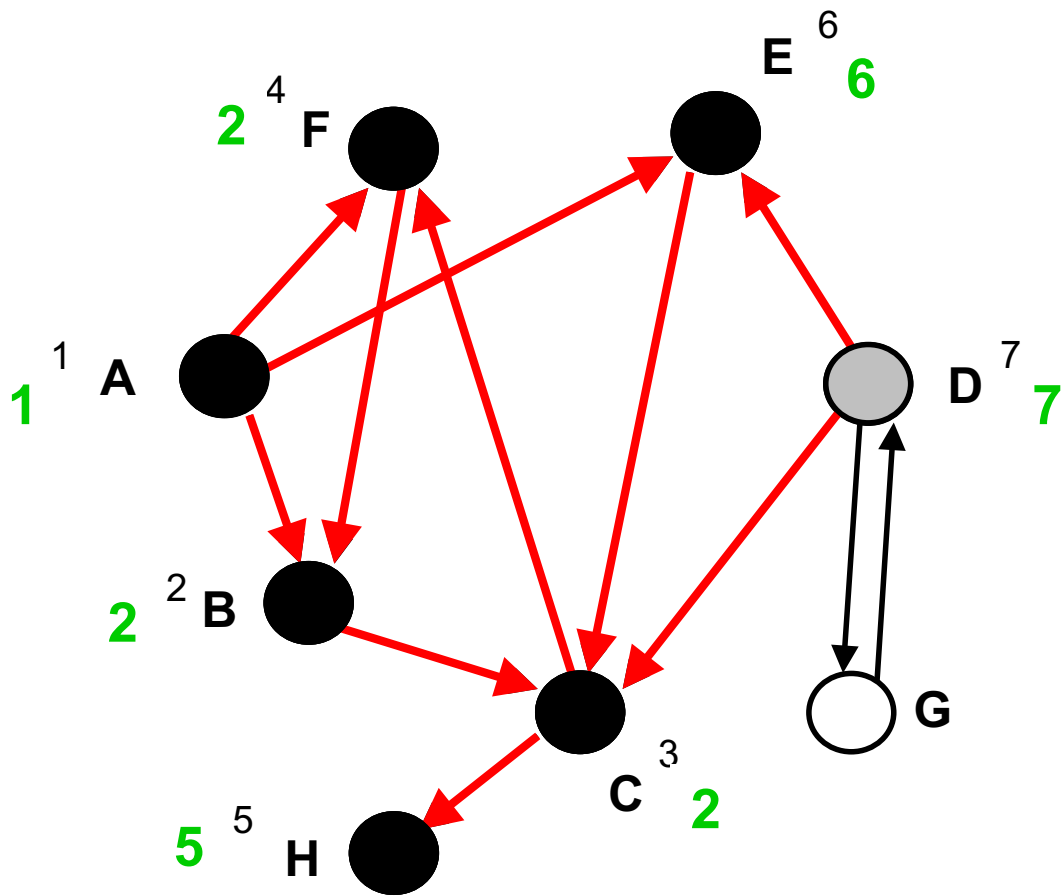
Exemplo



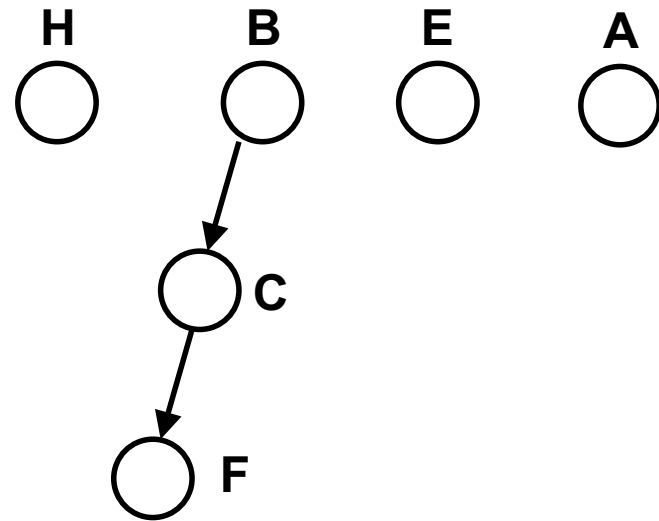
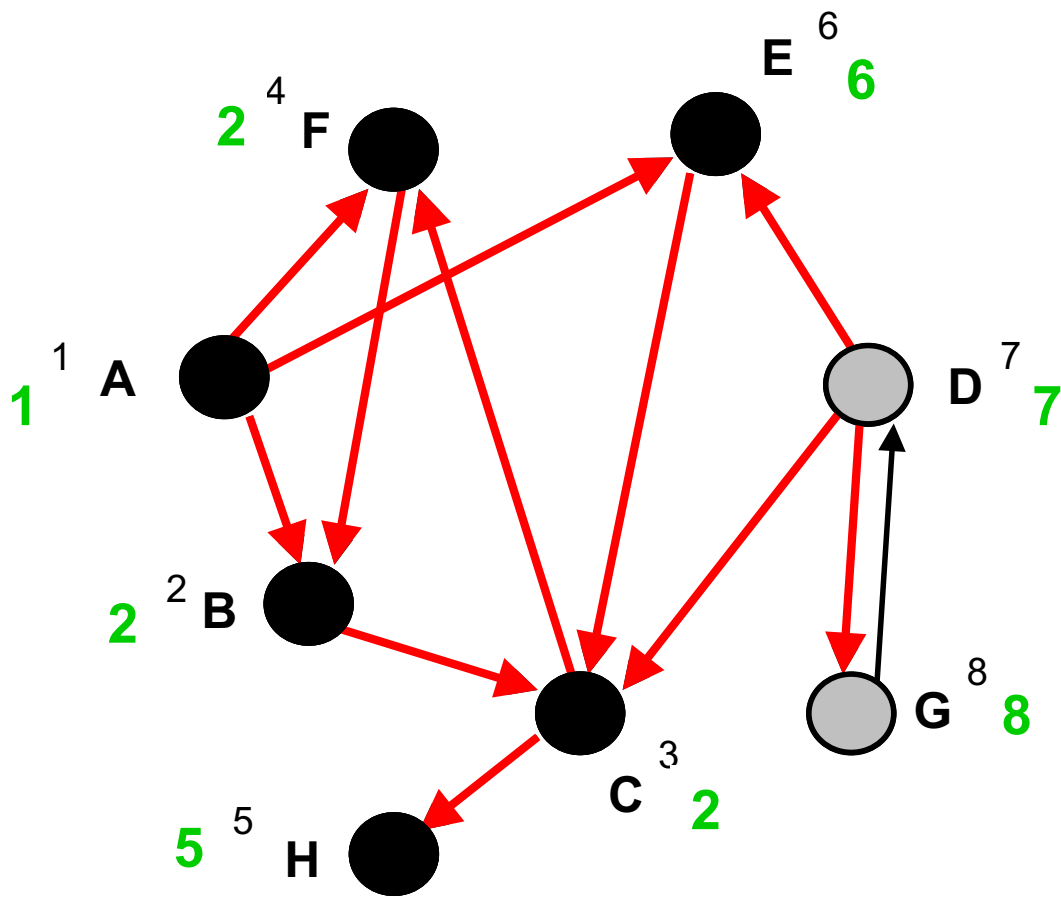
Exemplo



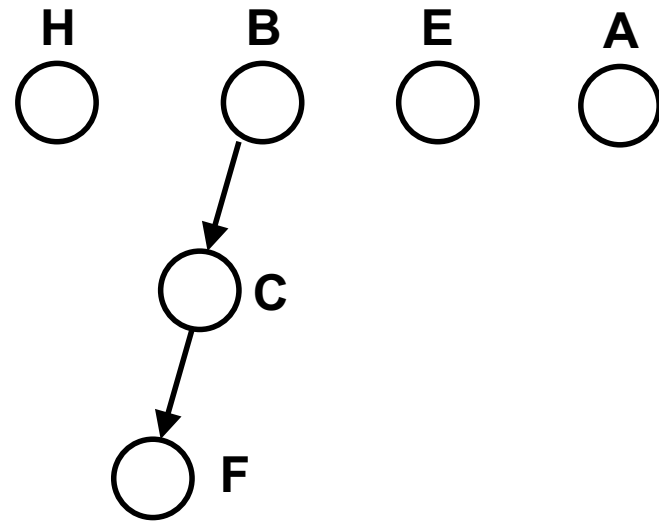
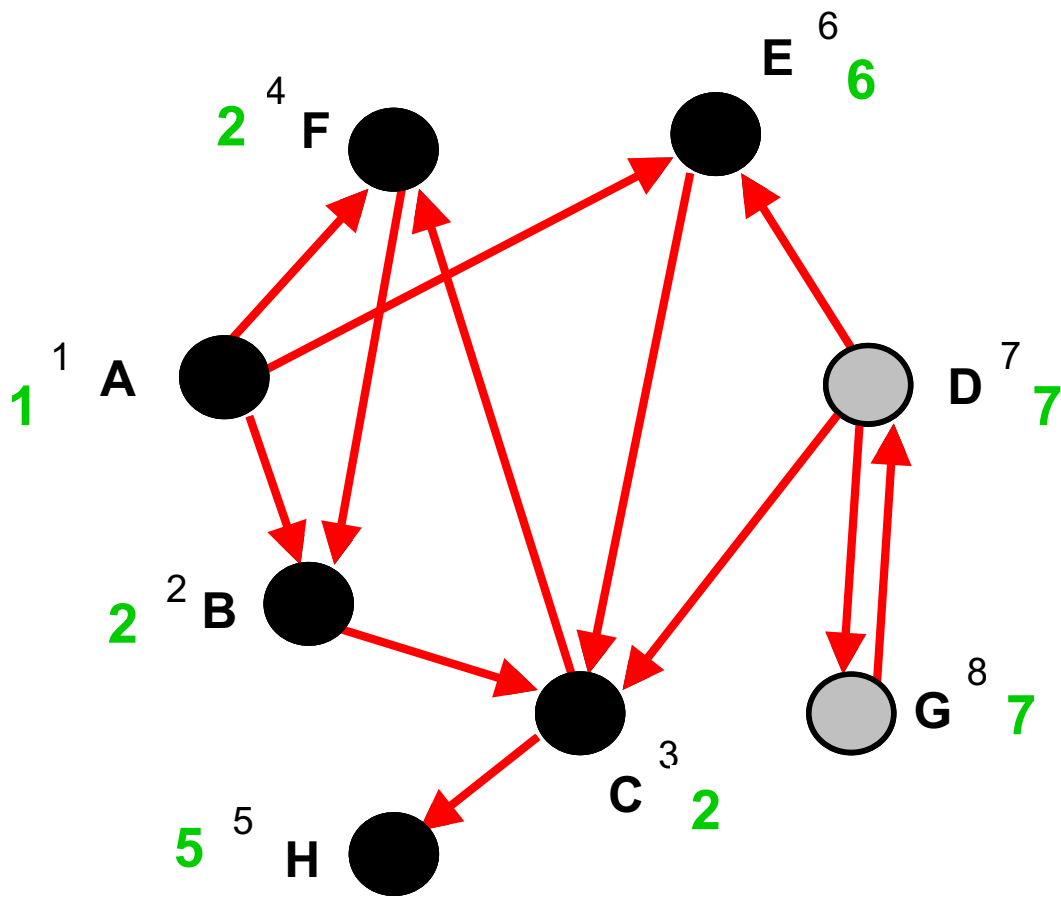
Exemplo



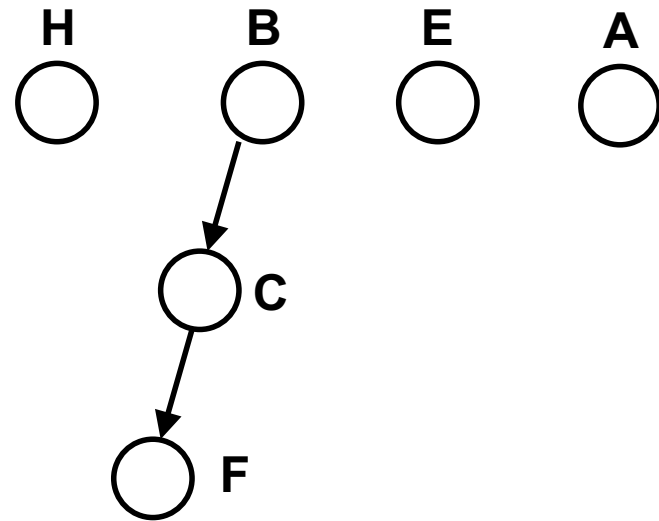
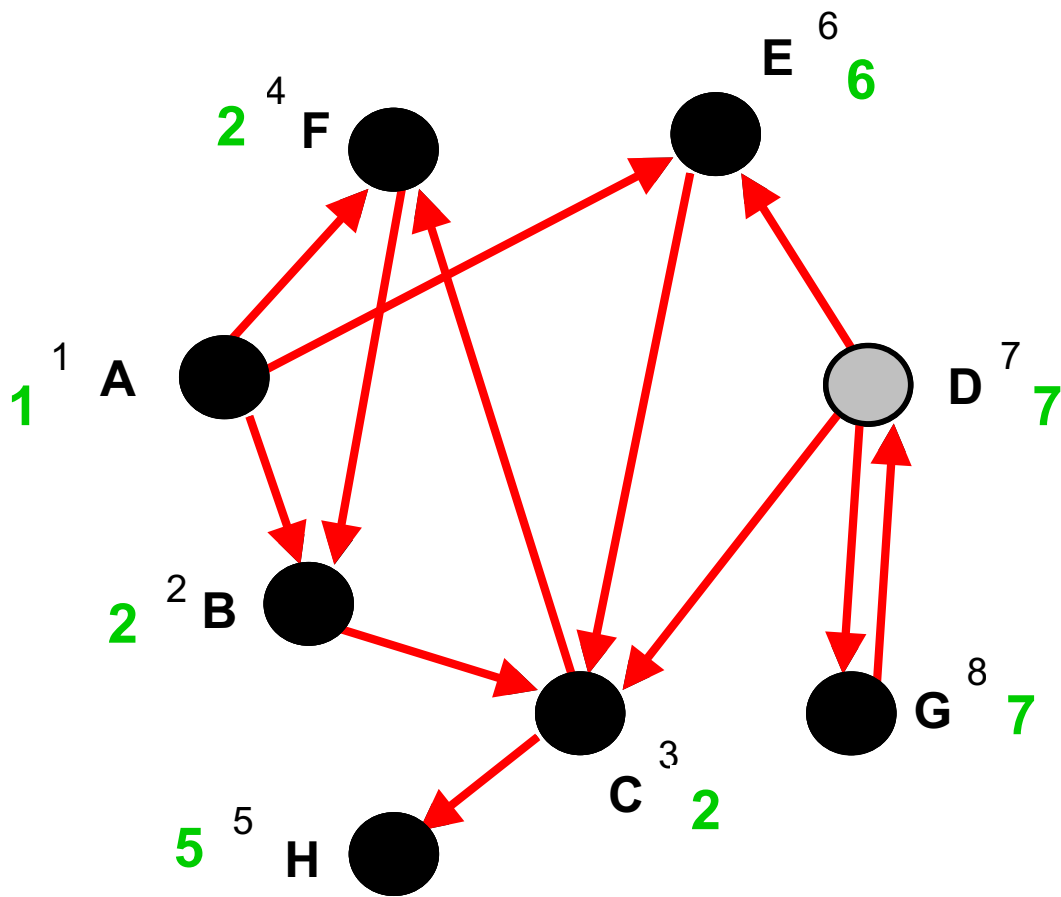
Exemplo



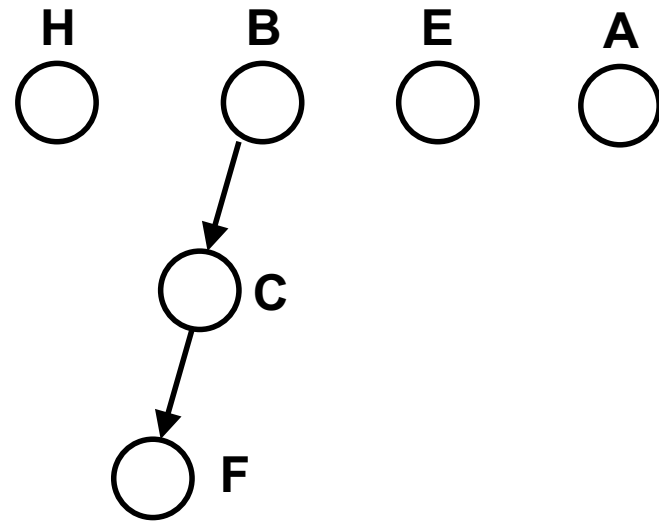
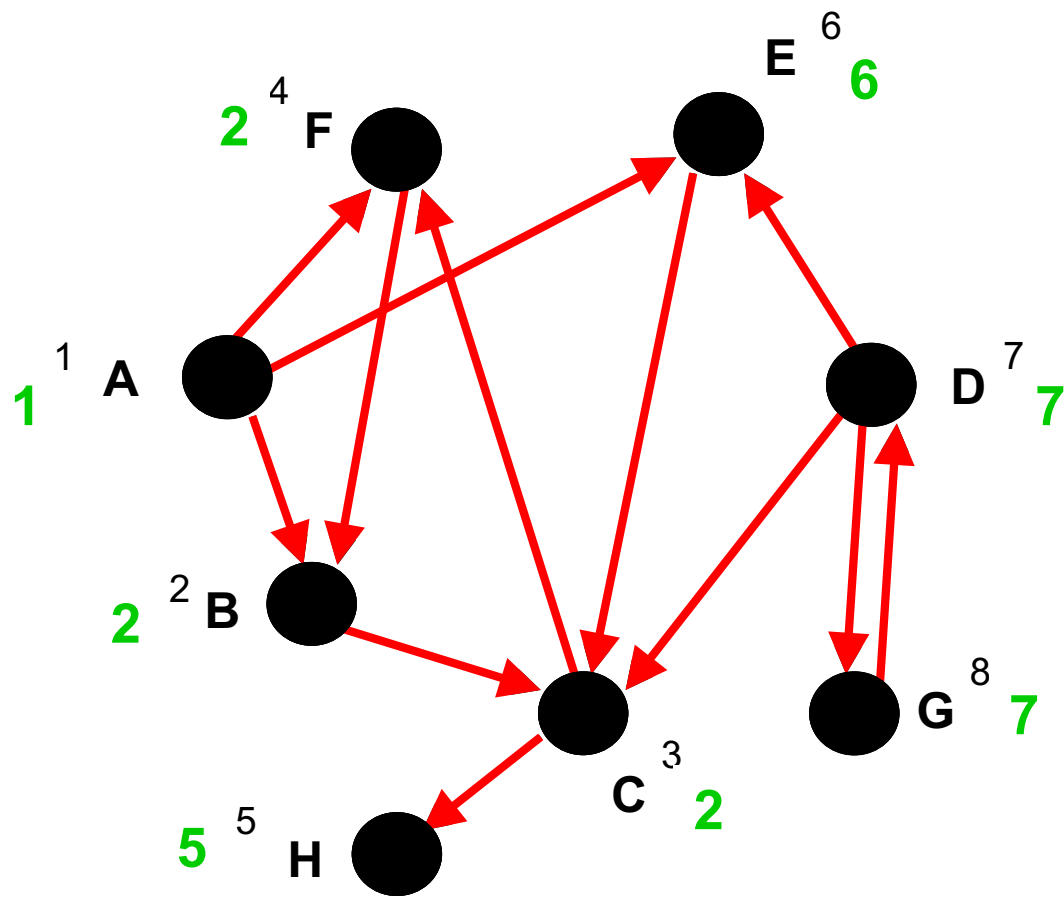
Exemplo



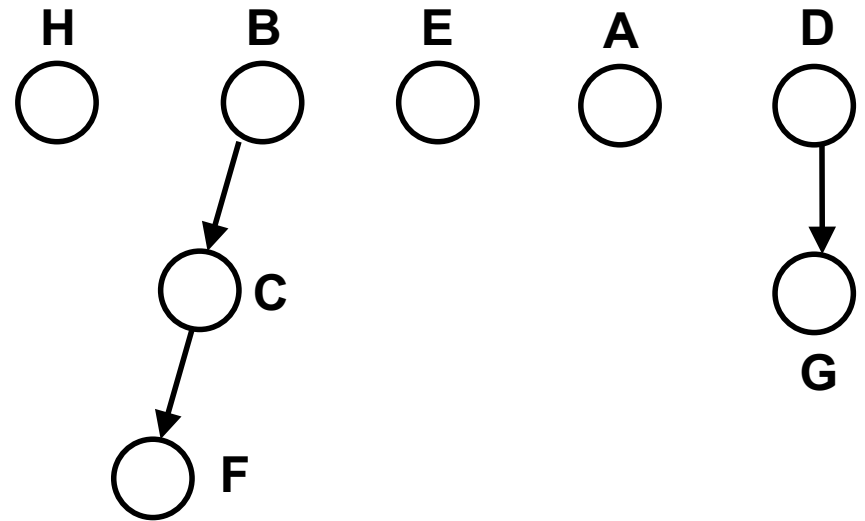
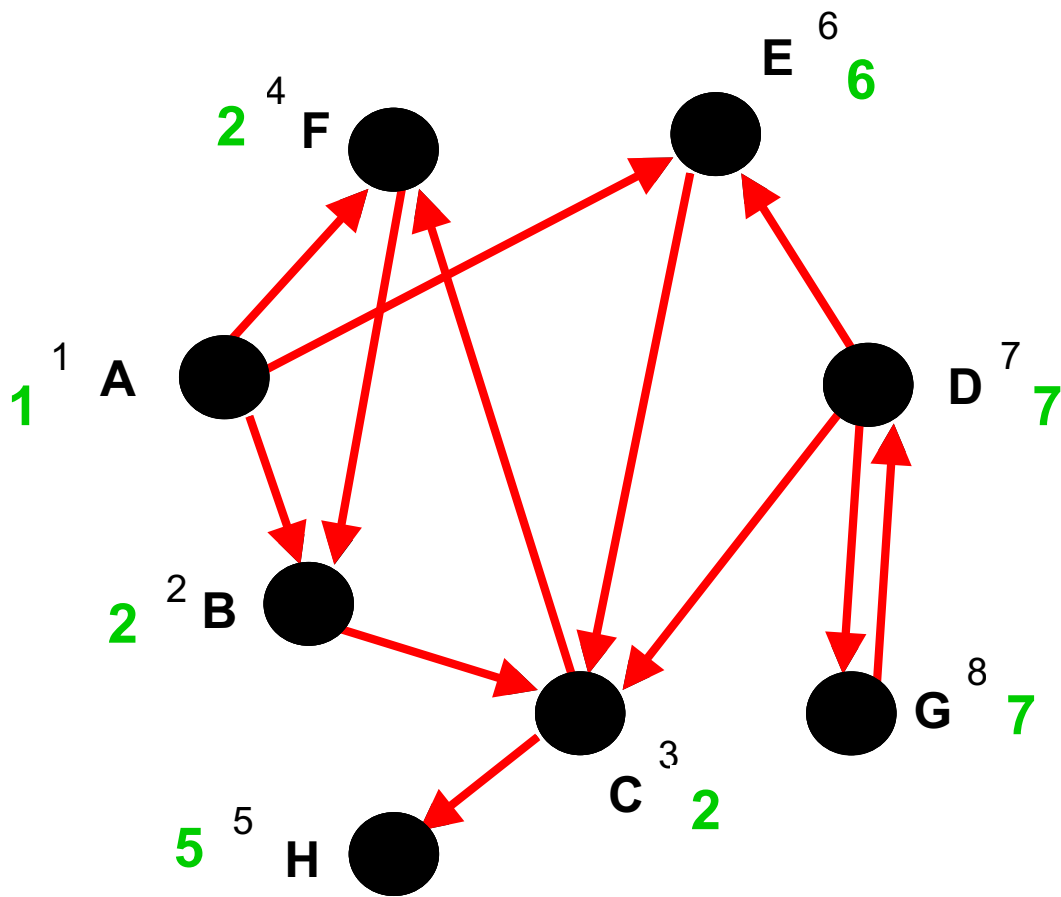
Exemplo



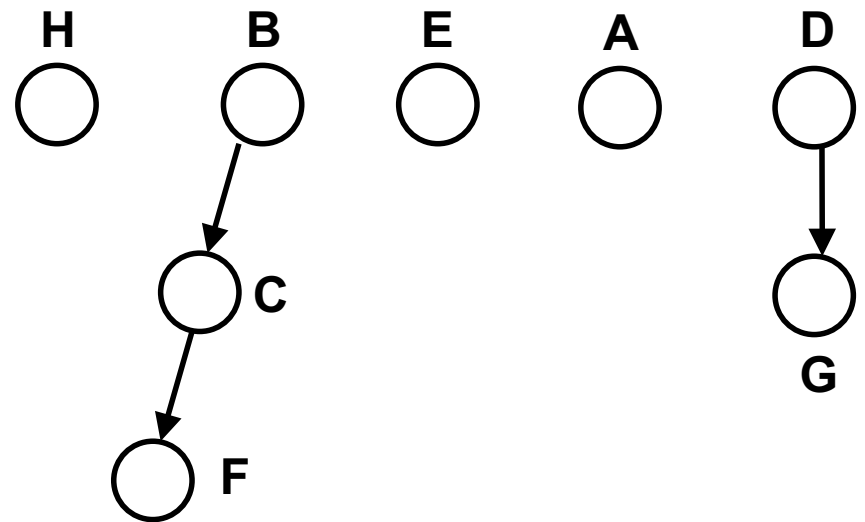
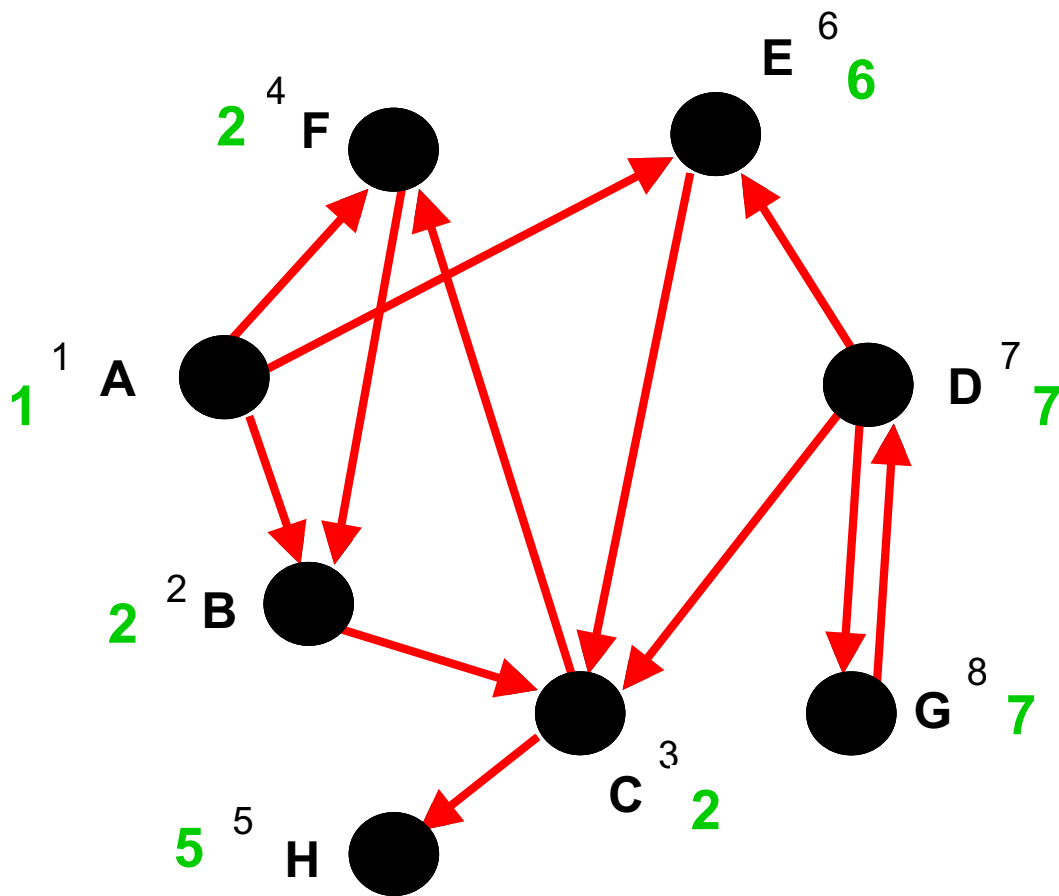
Exemplo



Exemplo

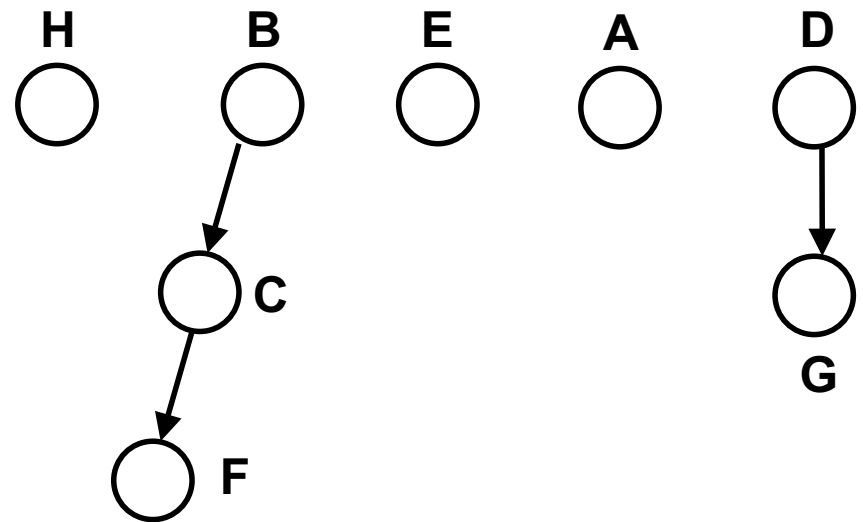
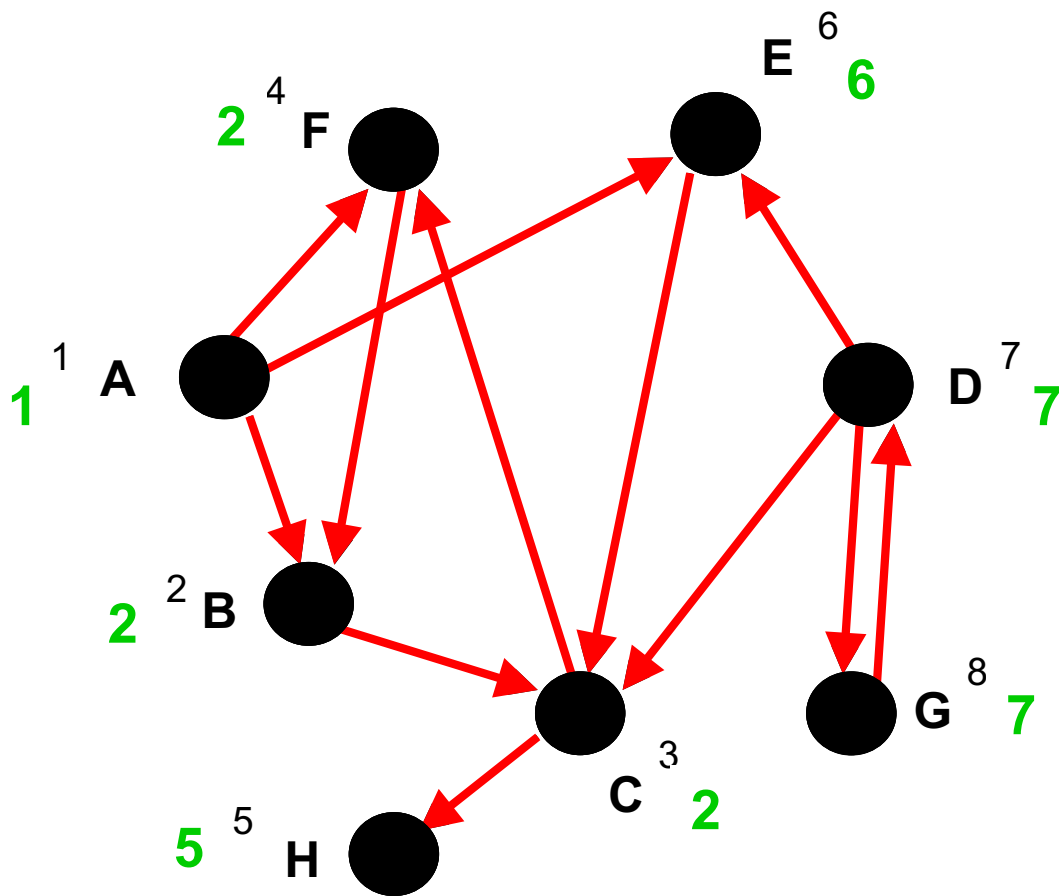


Exemplo



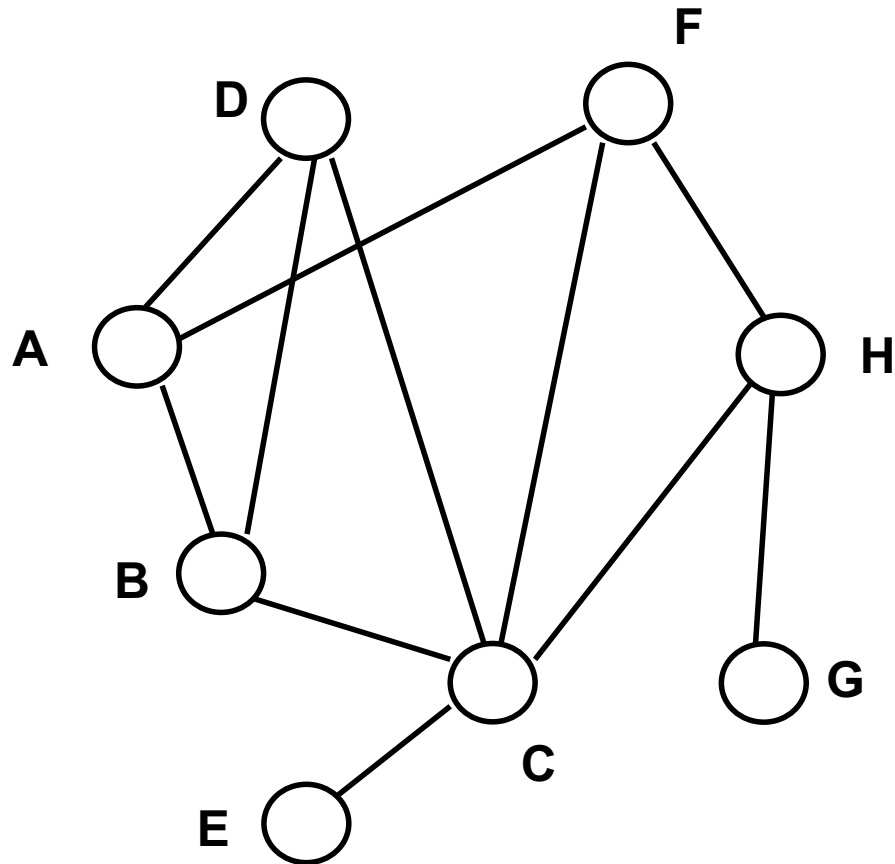
- Importante: nem todos os vértices de uma mesma componente terminam com o mesmo valor.

Exemplo

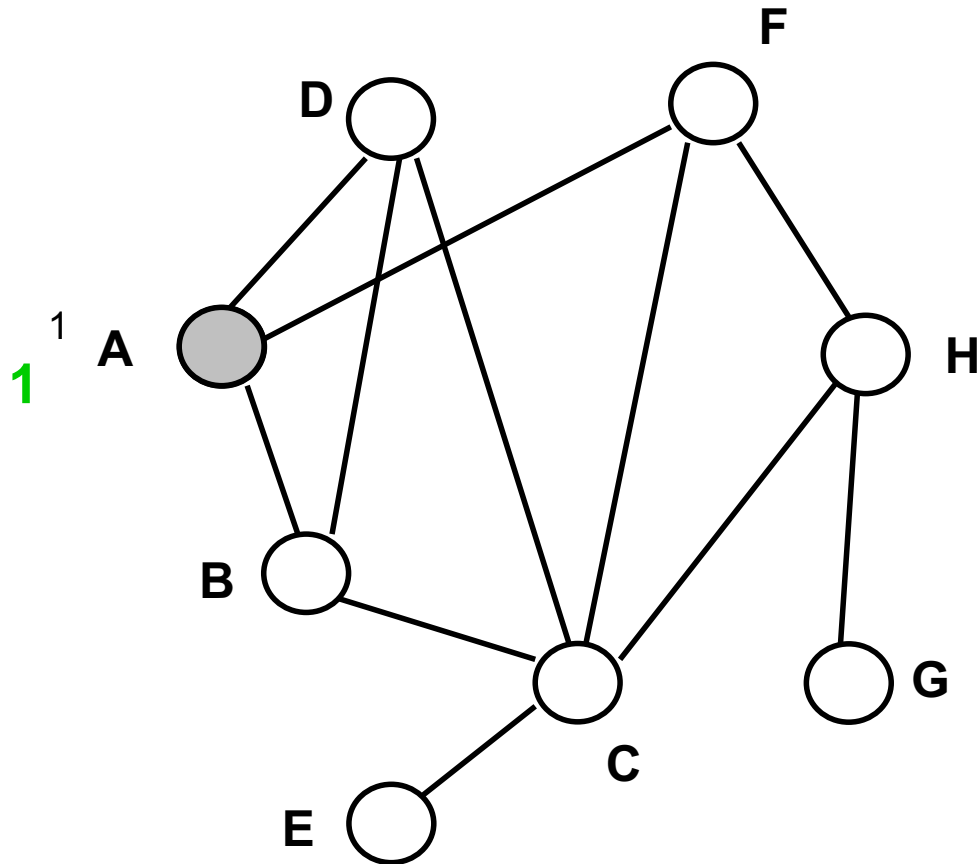


- Importante: nem todos os vértices de uma mesma componente terminam com o mesmo valor.
- Exemplo: acrescenta um arco $\langle C, A \rangle$ nesse mesmo digrafo, e visite $\langle C, F \rangle$ antes.

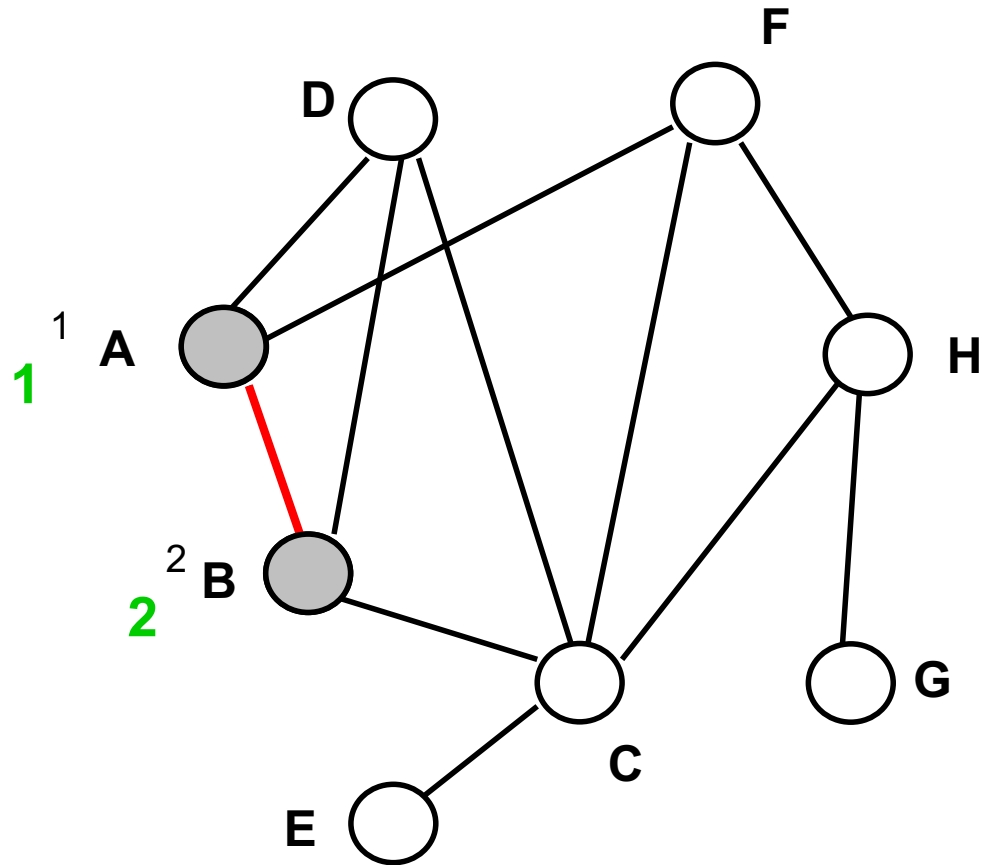
Exemplo



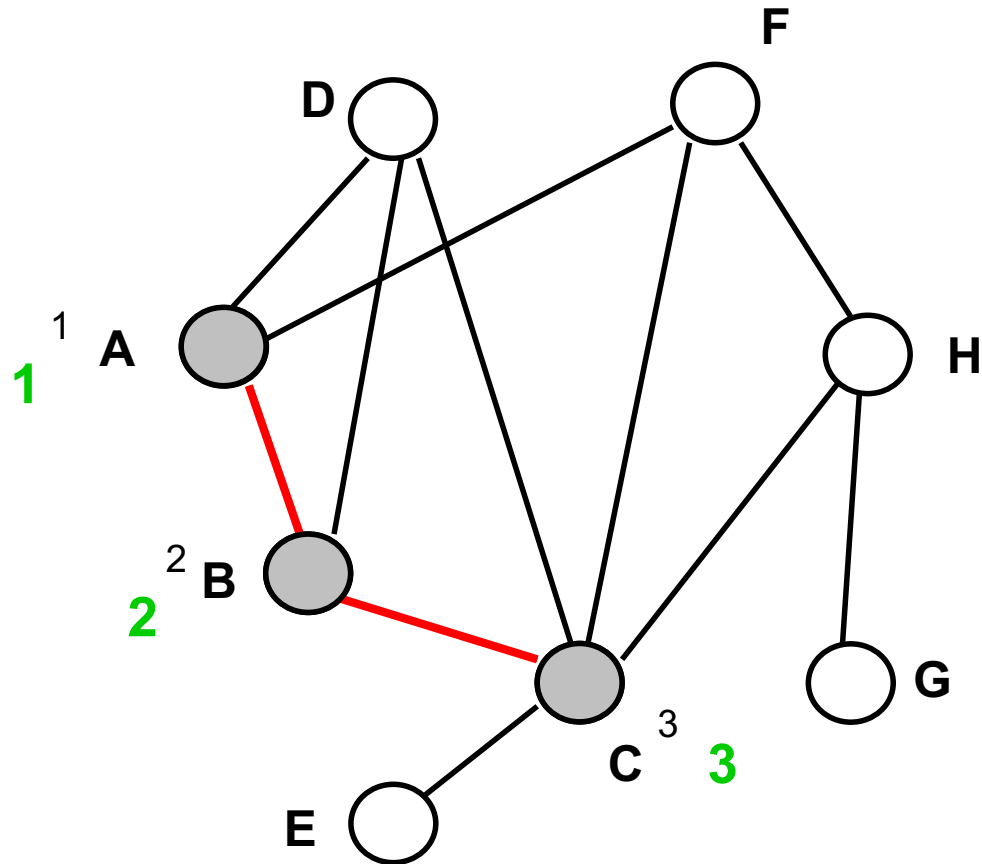
Exemplo



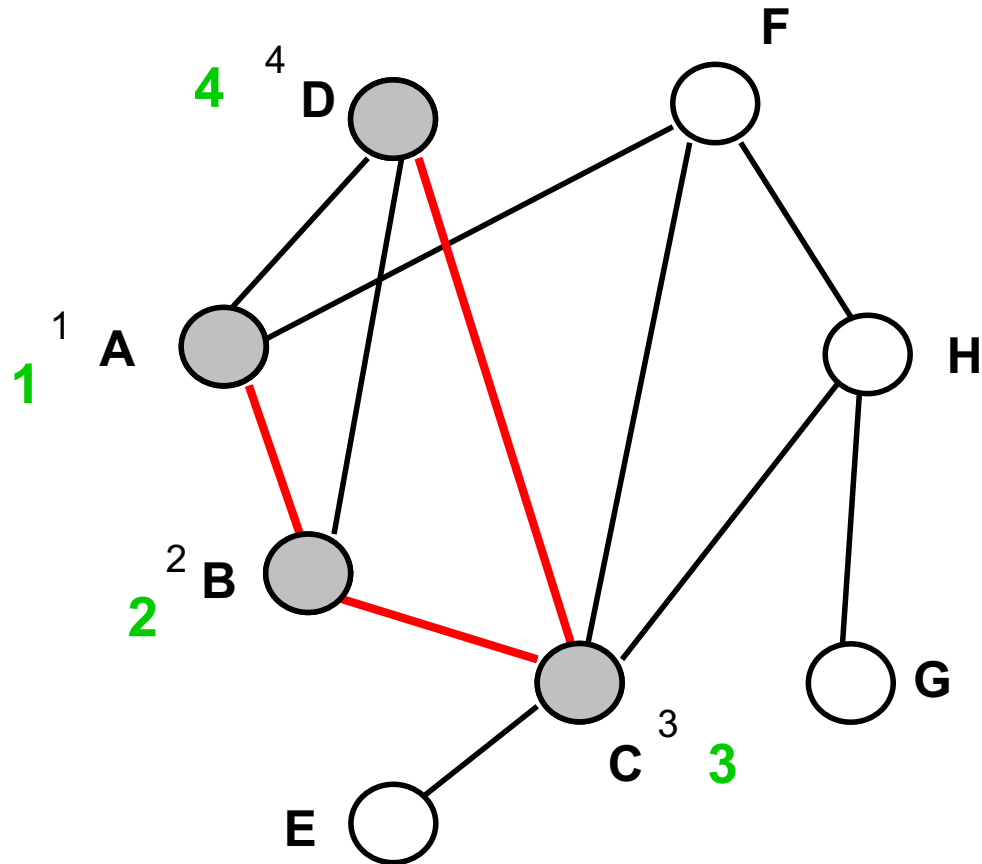
Exemplo



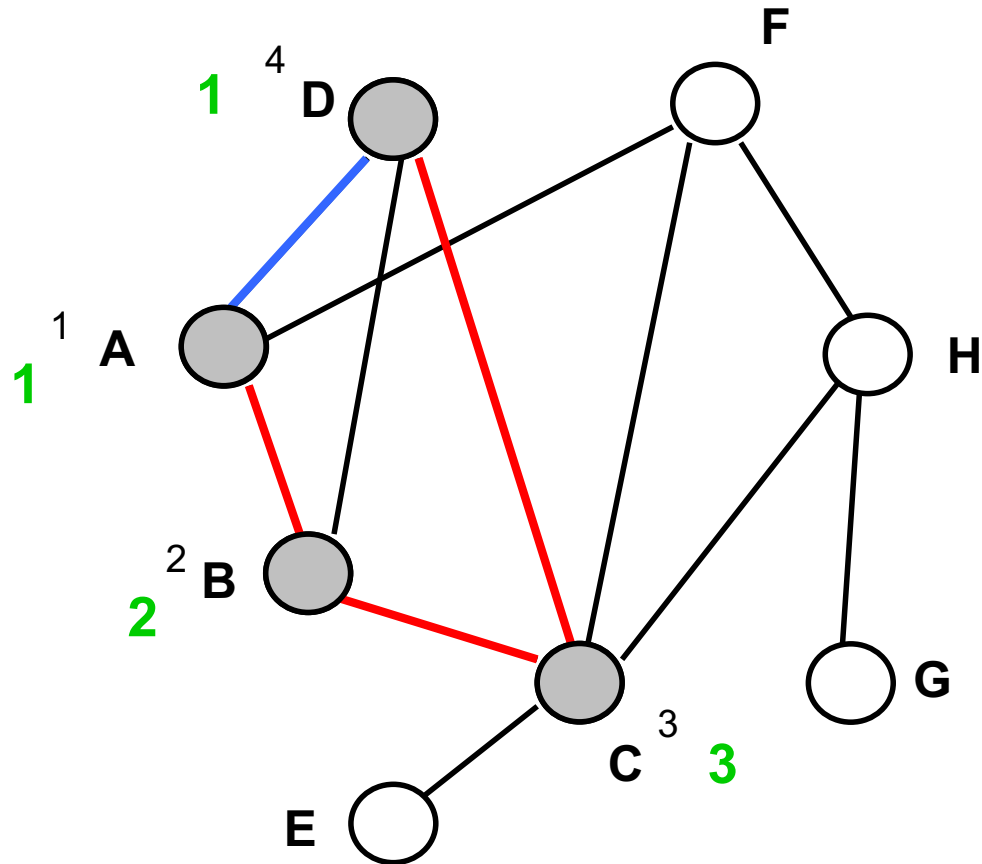
Exemplo



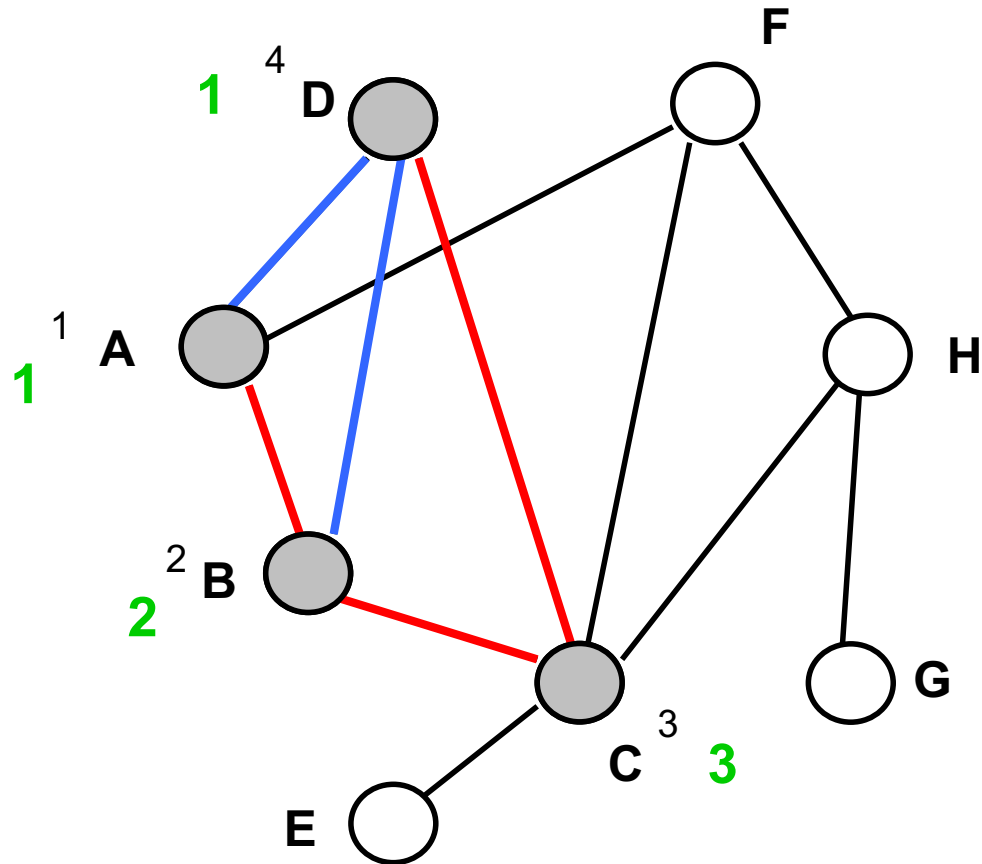
Exemplo



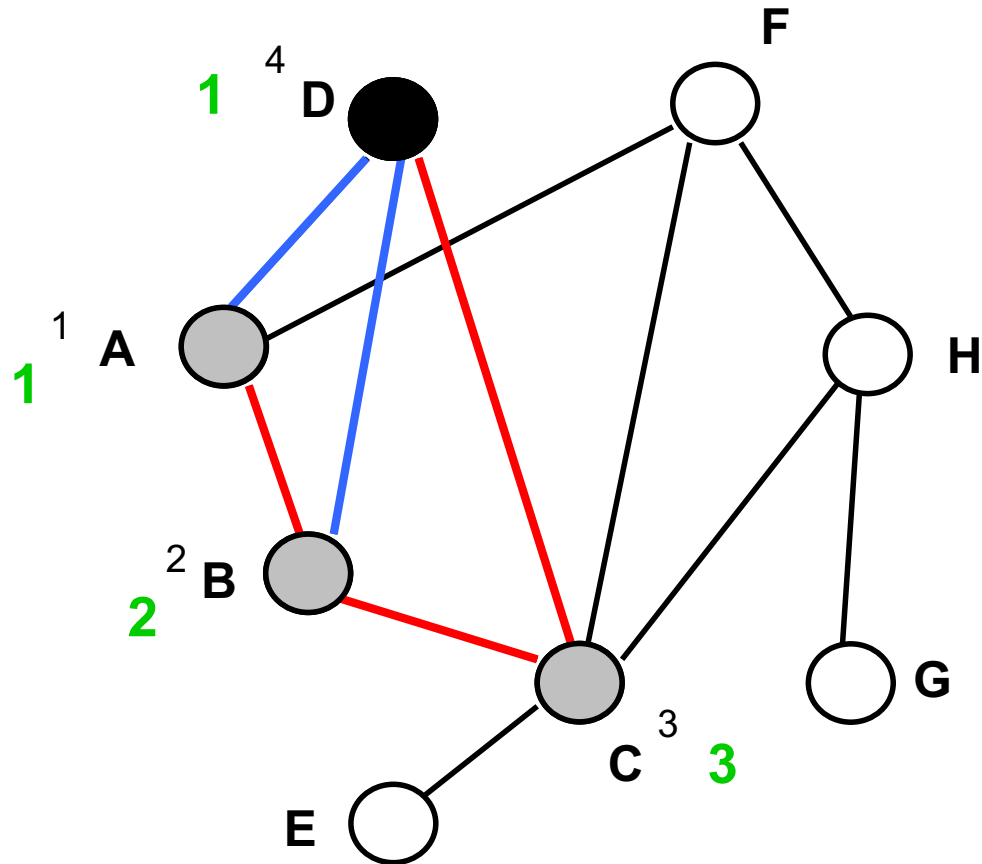
Exemplo



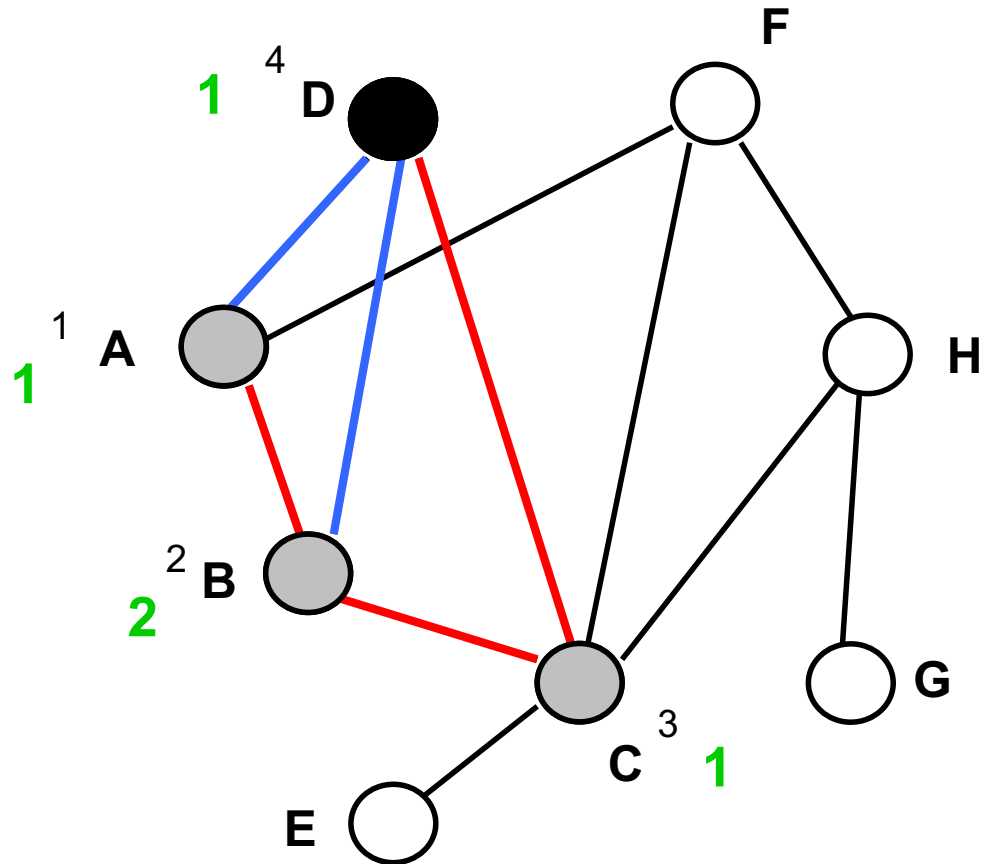
Exemplo



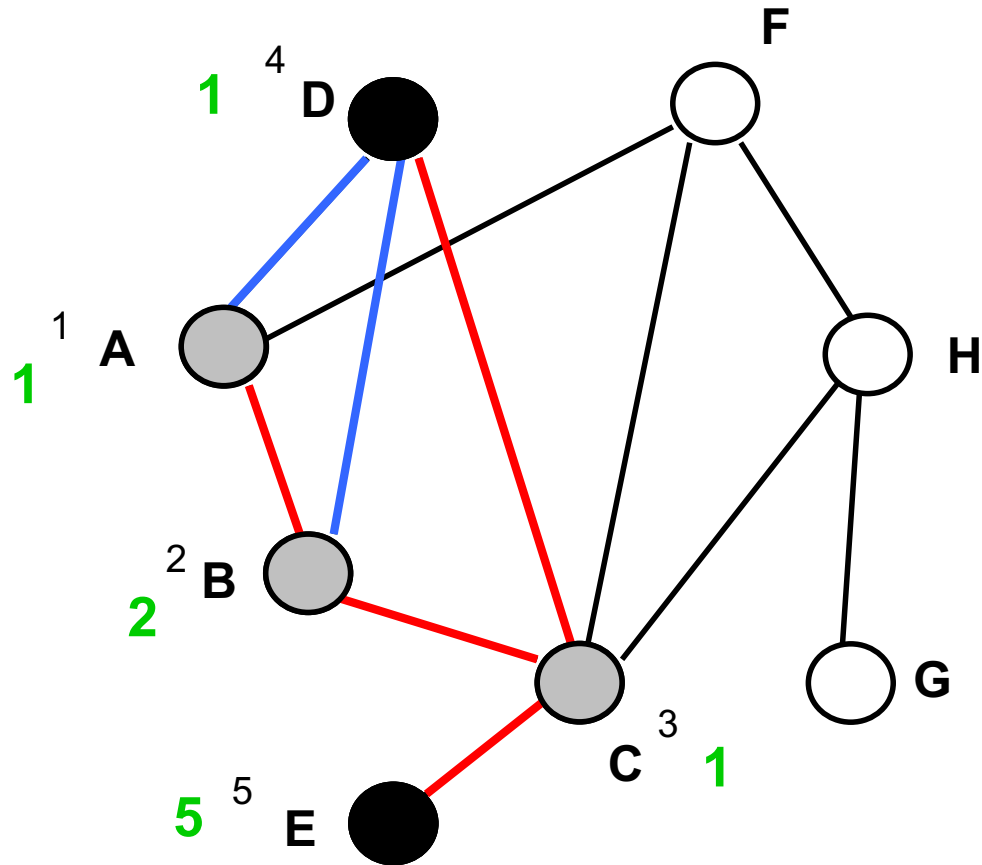
Exemplo



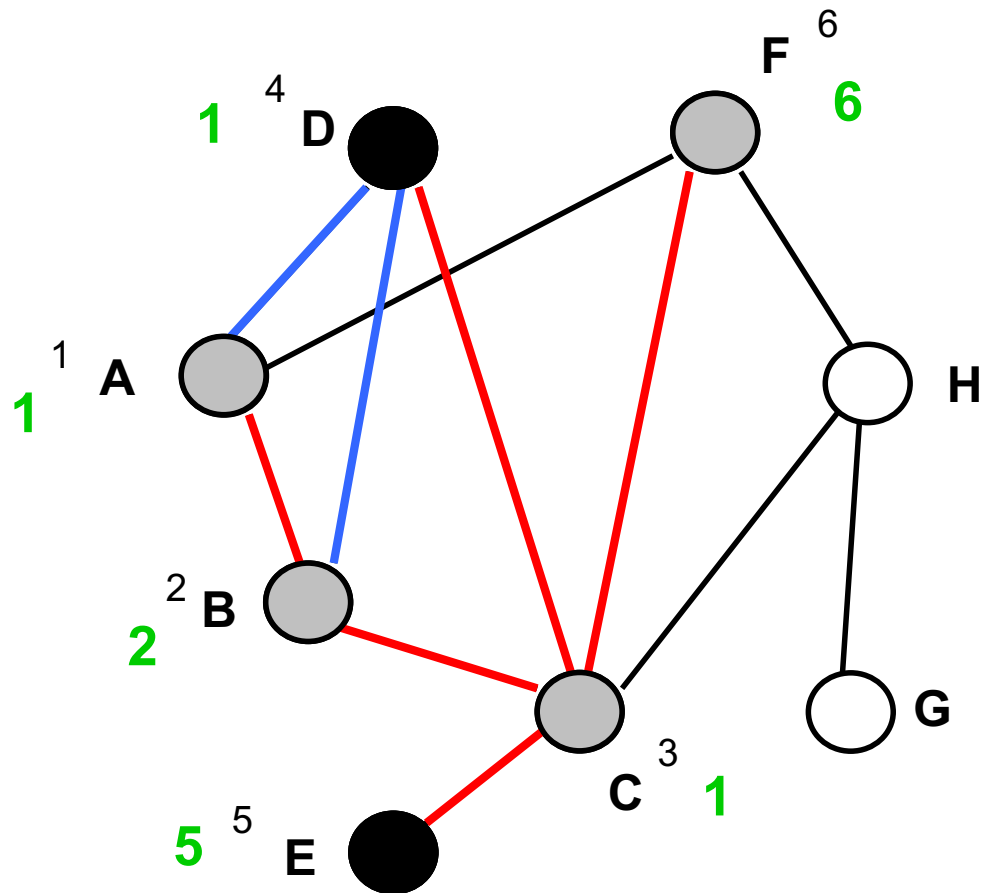
Exemplo



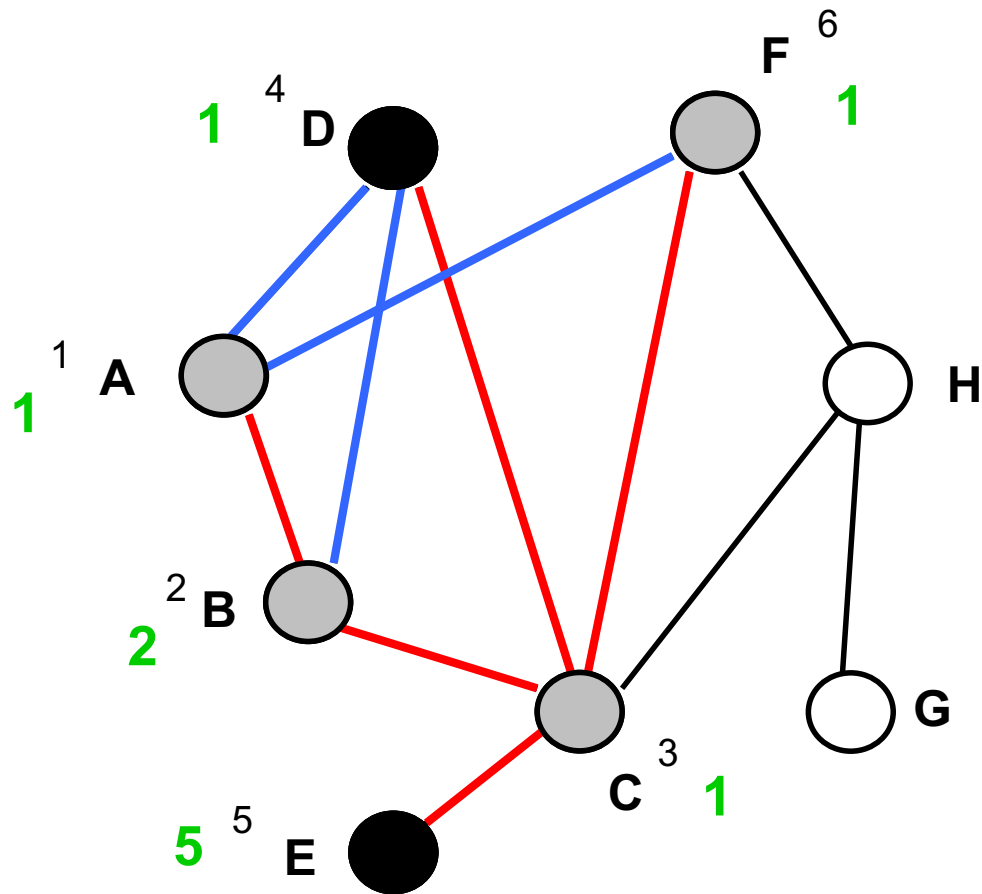
Exemplo



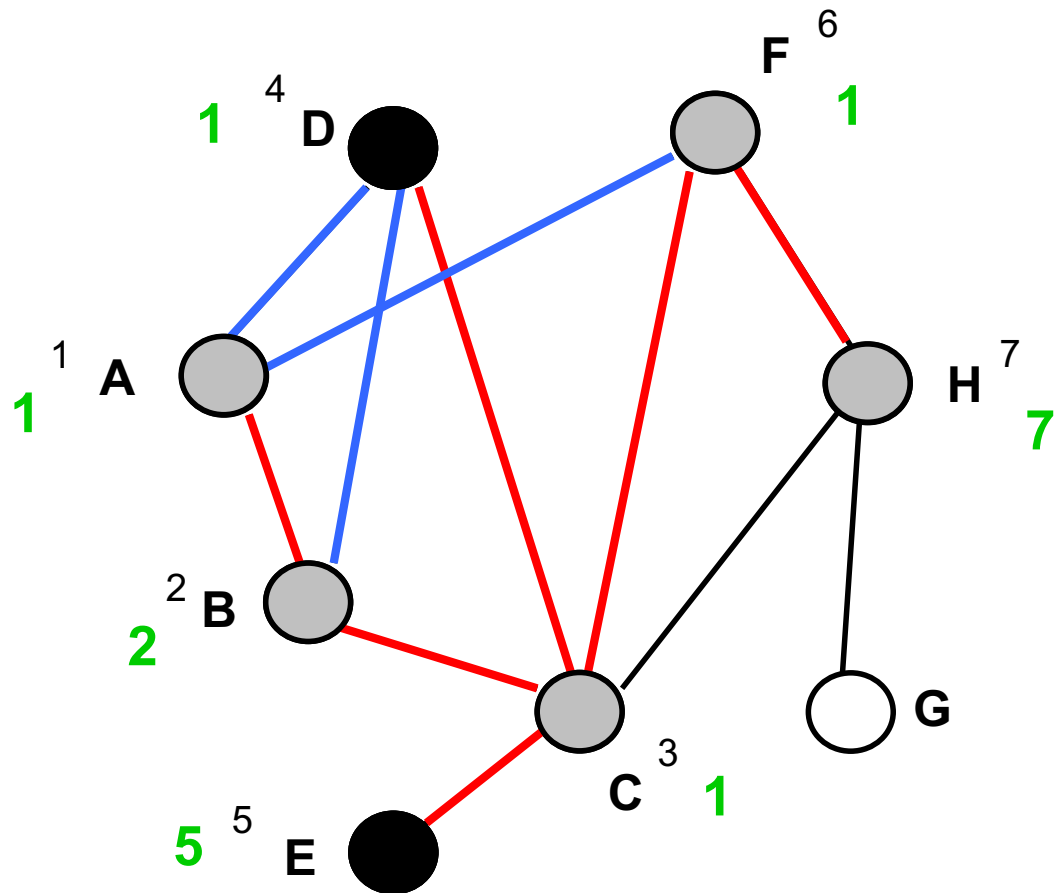
Exemplo



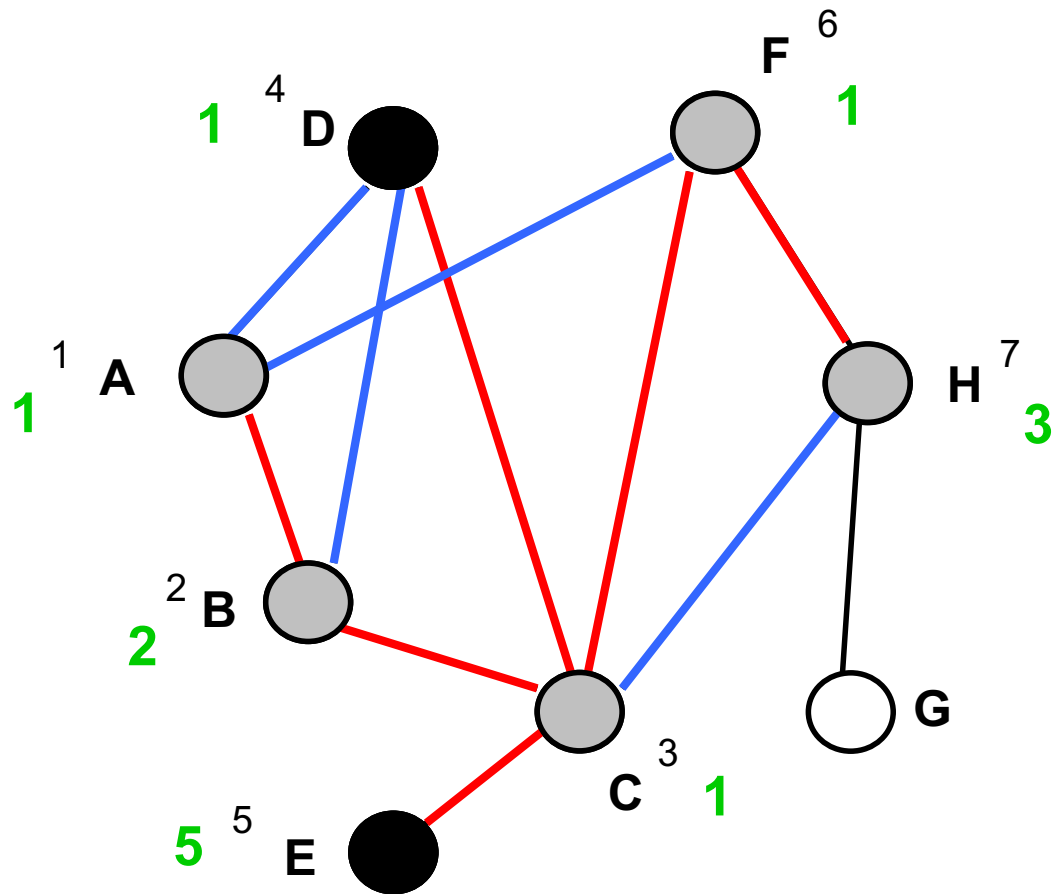
Exemplo



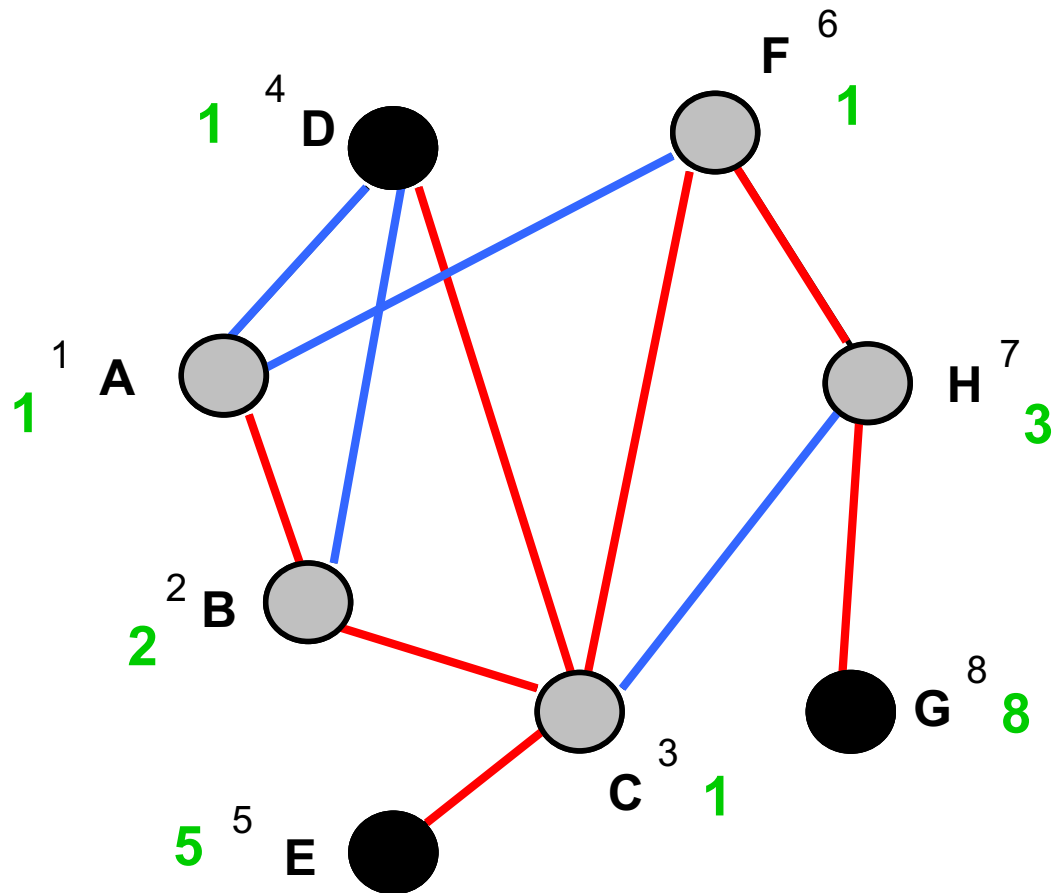
Exemplo



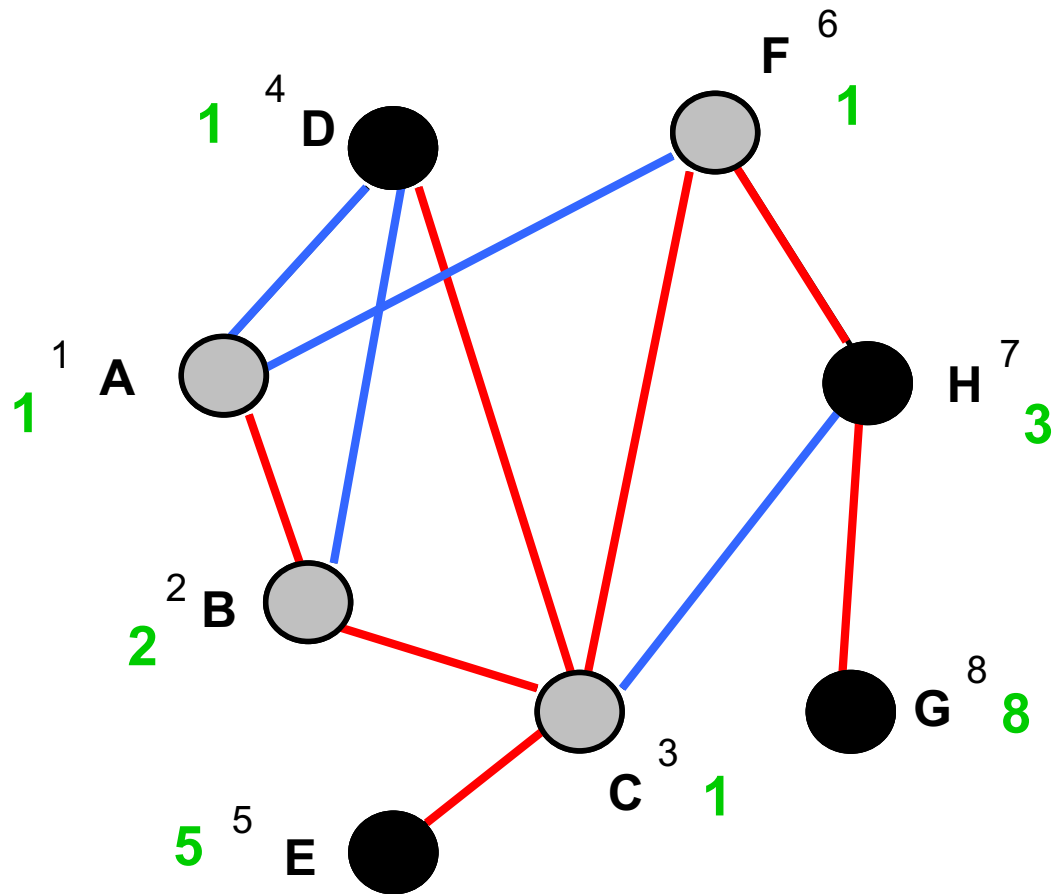
Exemplo



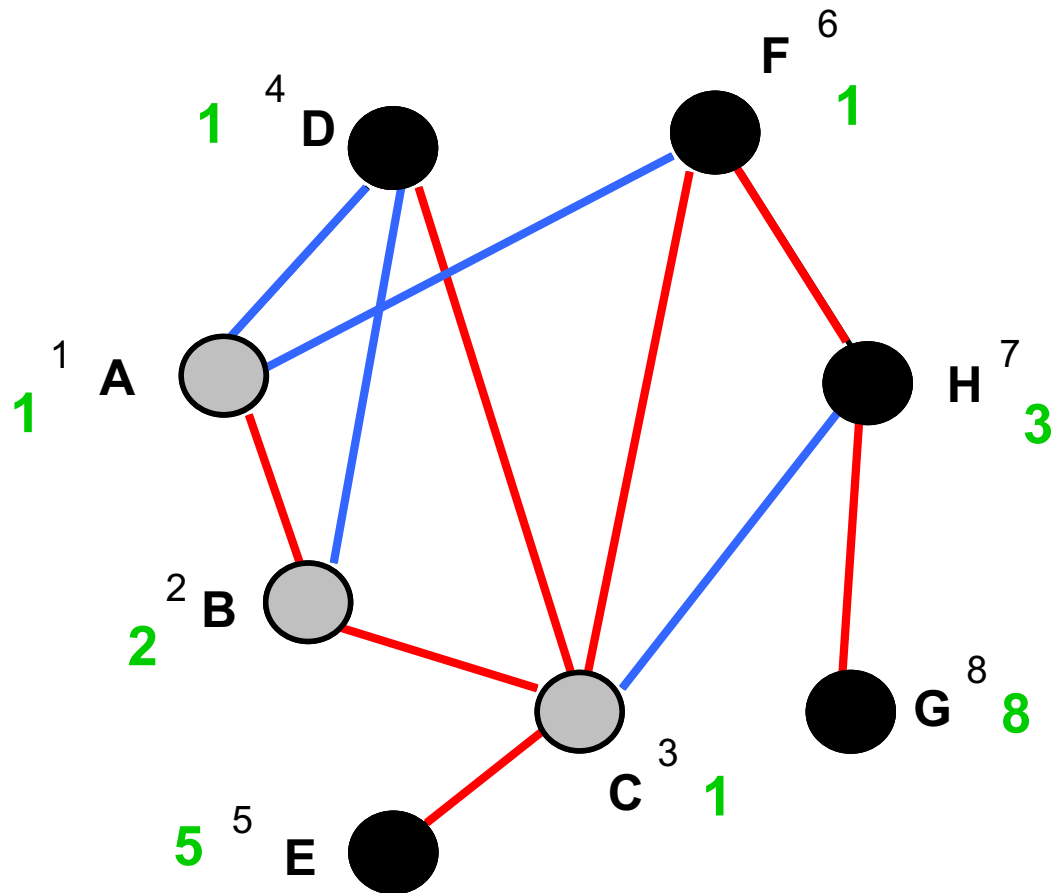
Exemplo



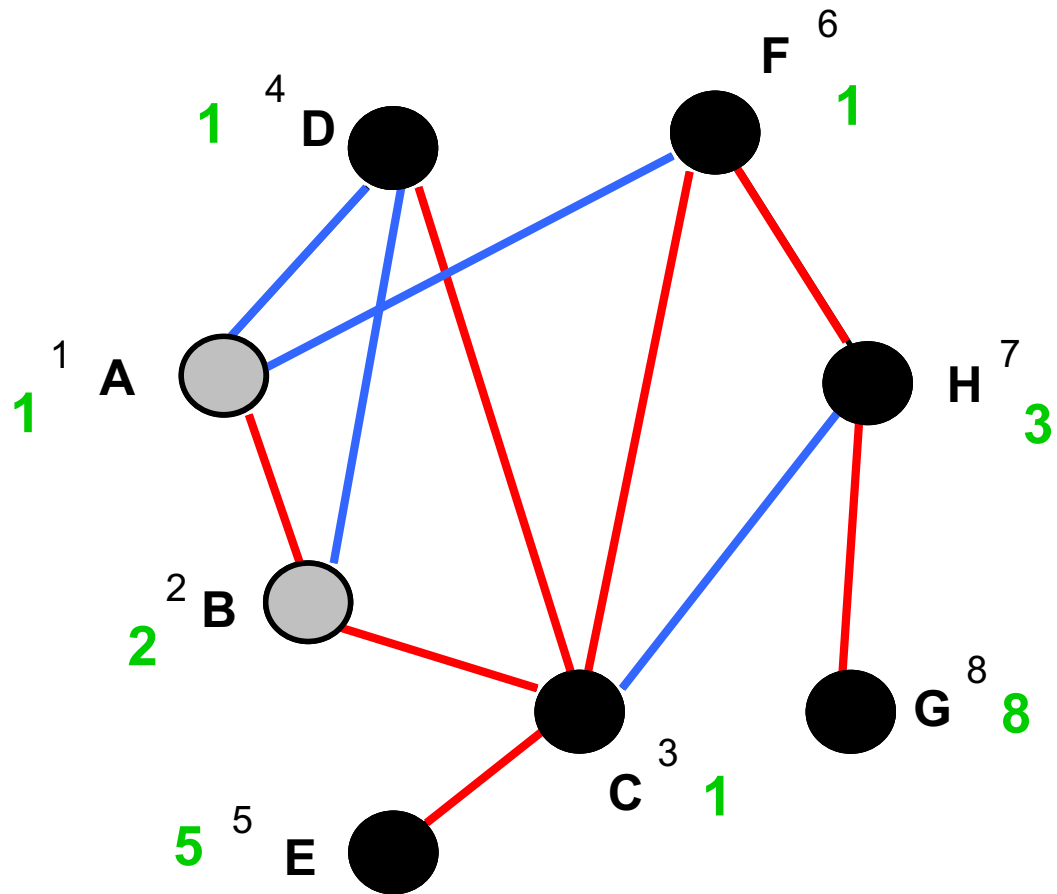
Exemplo



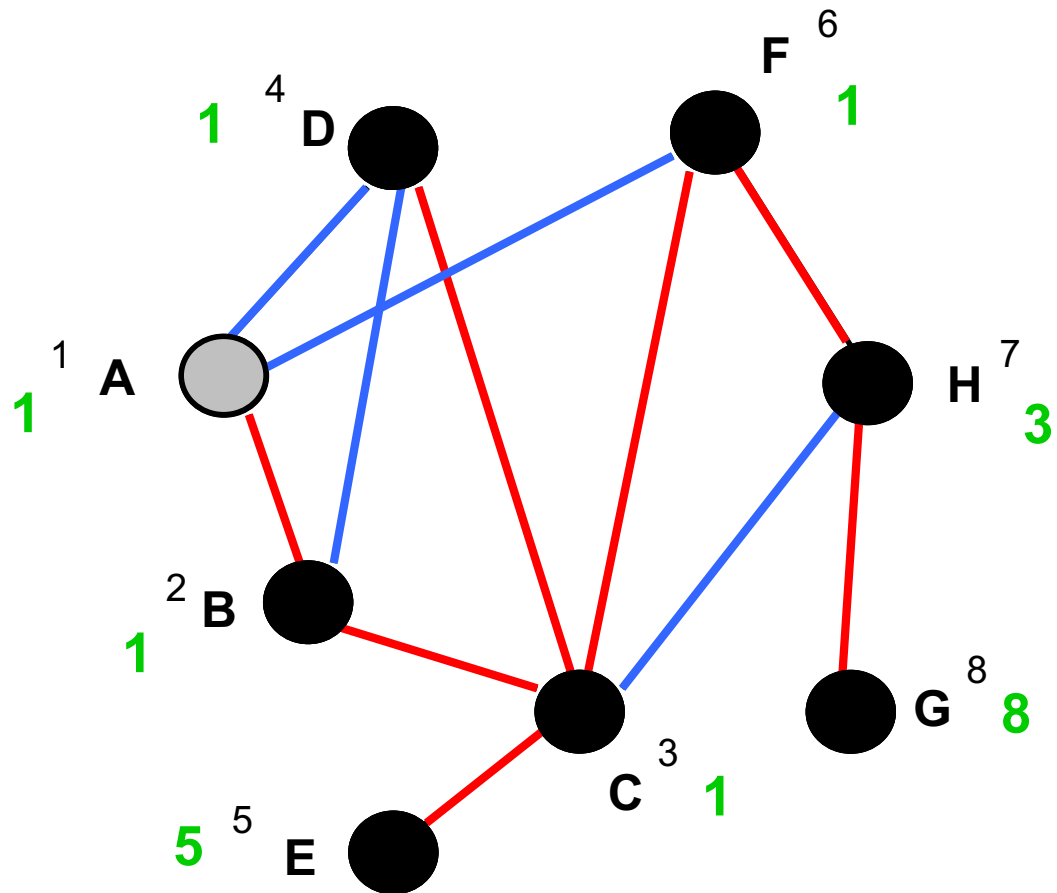
Exemplo



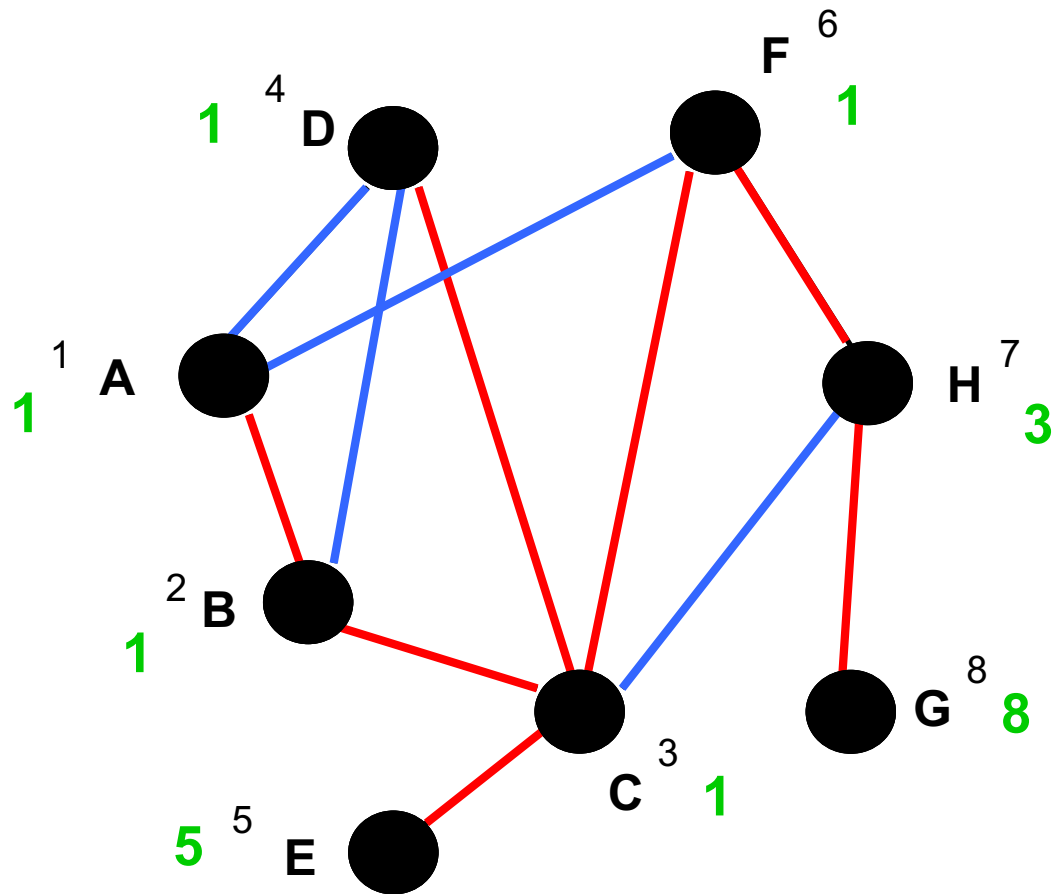
Exemplo



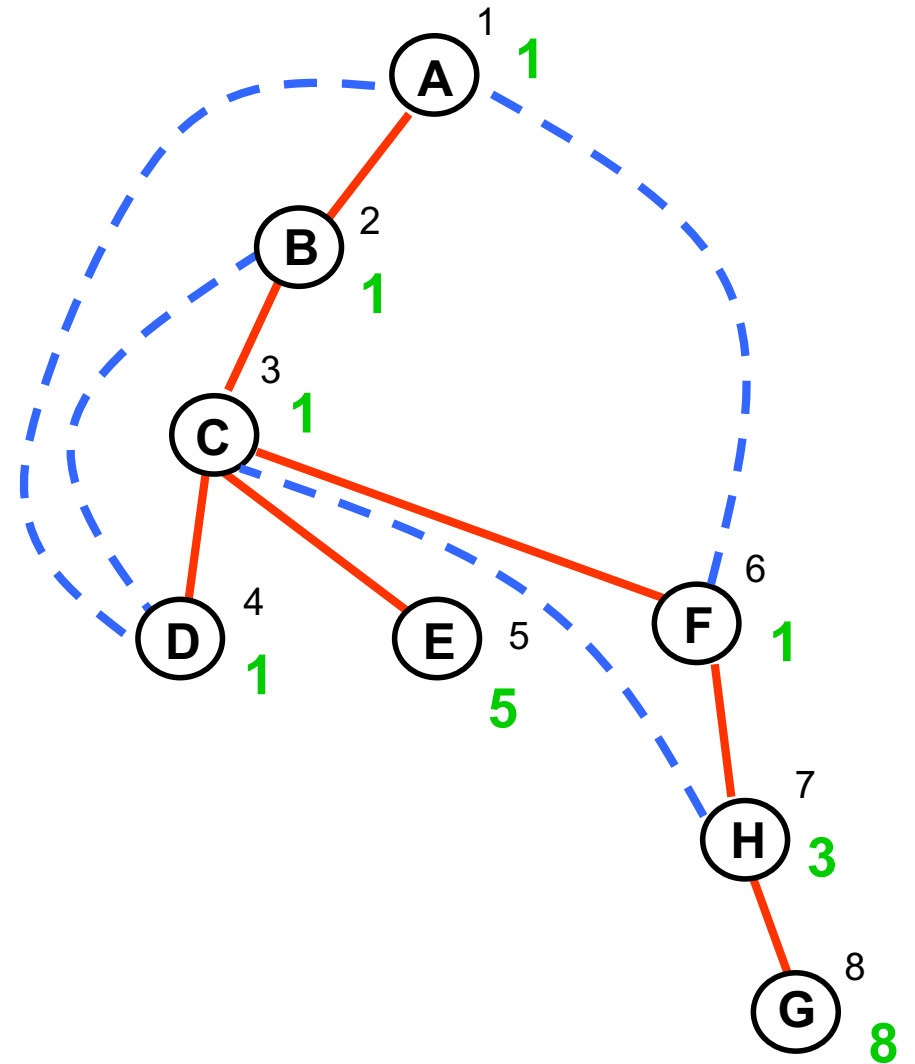
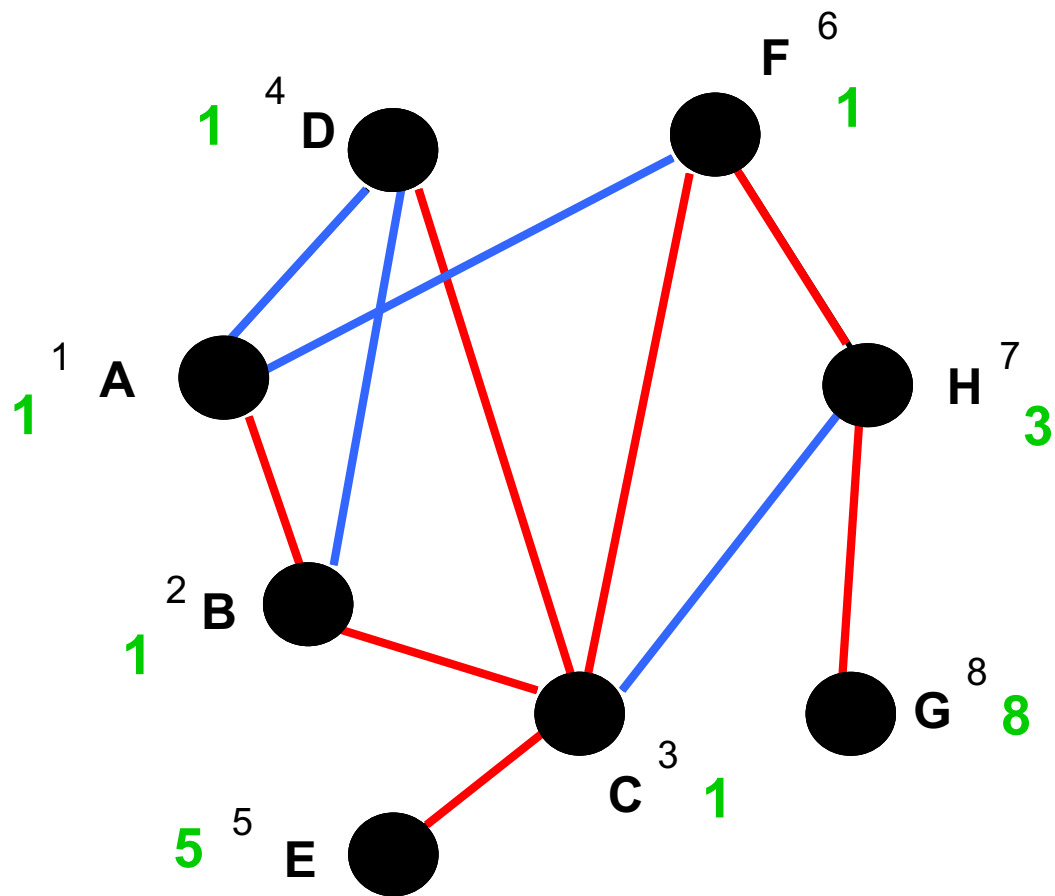
Exemplo



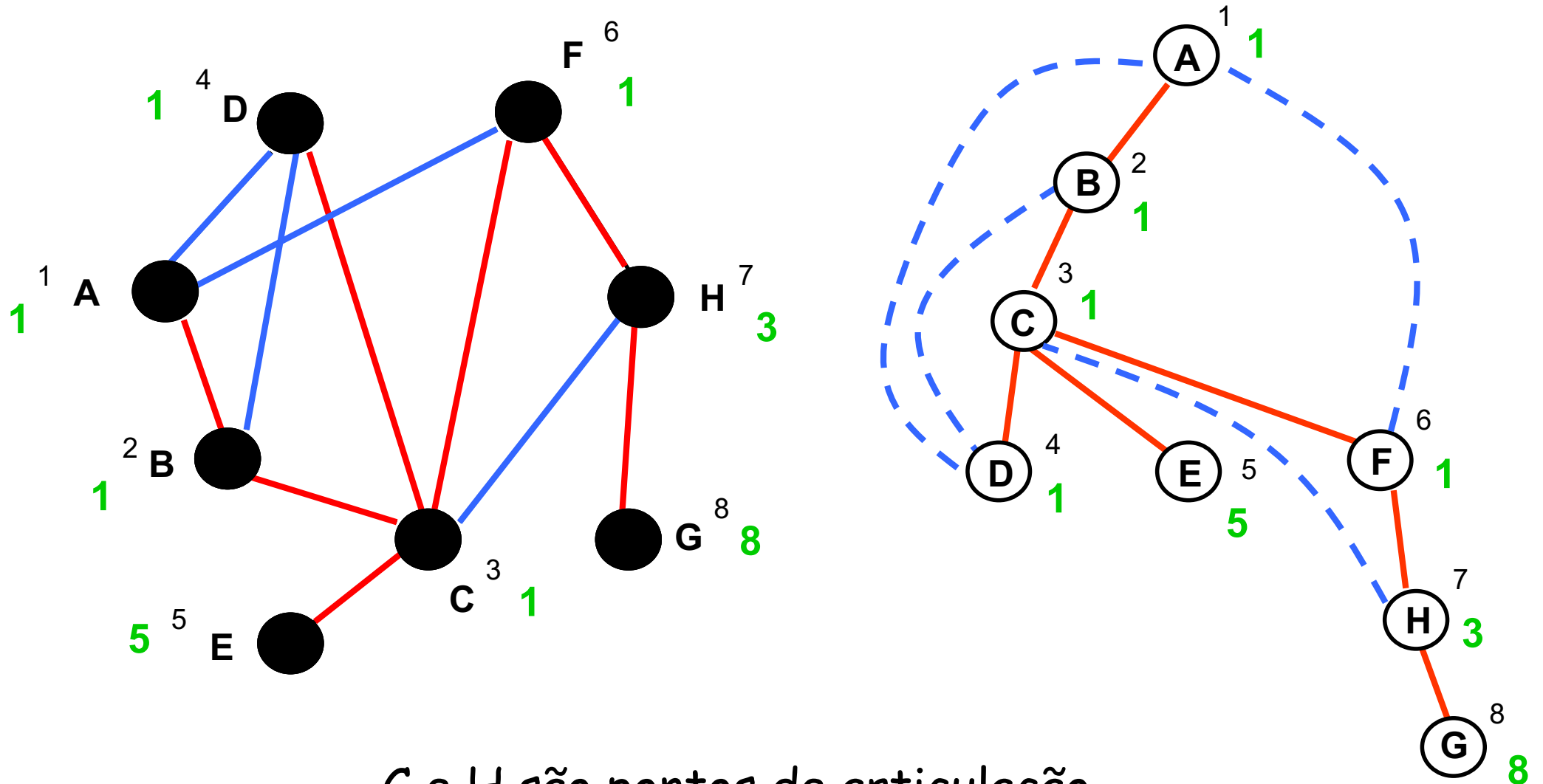
Exemplo



Exemplo

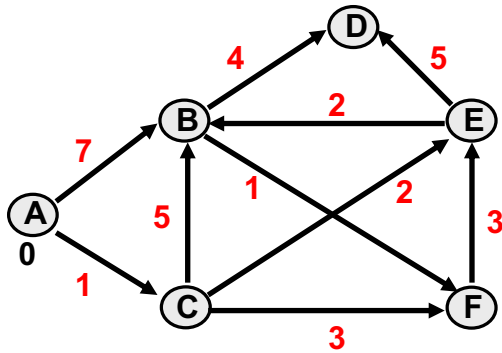


Exemplo



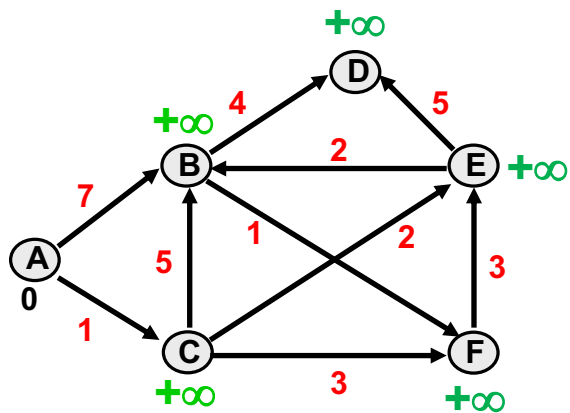
C e H são pontos de articulação

Exemplo do algoritmo de Dijkstra



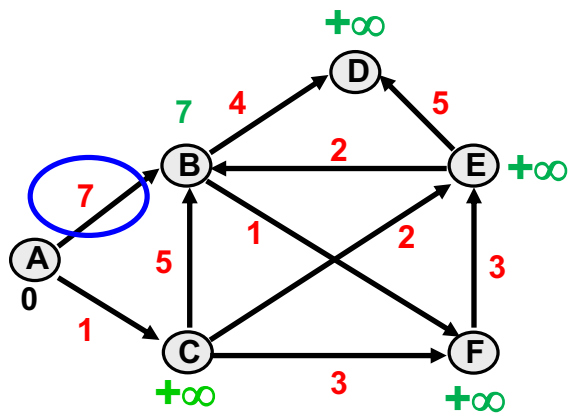
Exemplo do algoritmo de Dijkstra

Pesos dos arcos, distâncias provisórias, distâncias definitivas



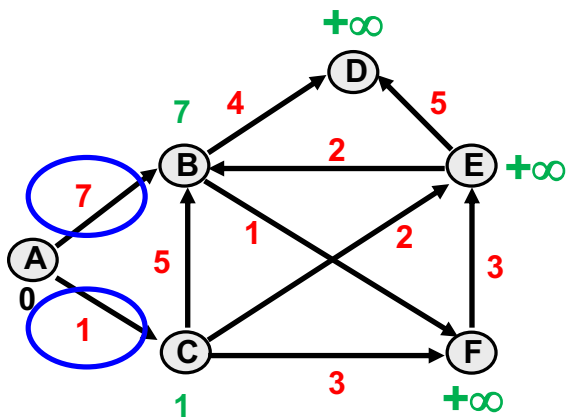
Exemplo do algoritmo de Dijkstra

Pesos dos arcos, distâncias provisórias, distâncias definitivas



Exemplo do algoritmo de Dijkstra

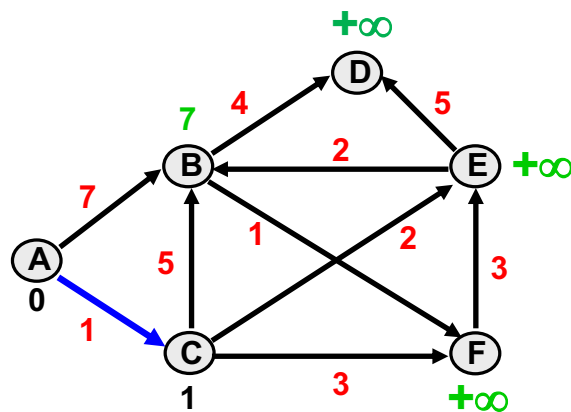
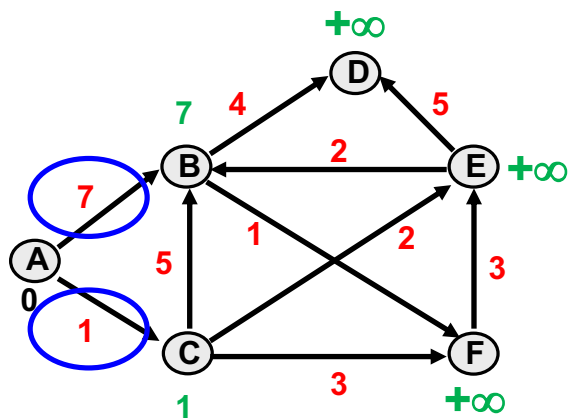
Pesos dos arcos, distâncias provisórias, distâncias definitivas



Exemplo do algoritmo de Dijkstra

Pesos dos arcos, distâncias provisórias, distâncias definitivas

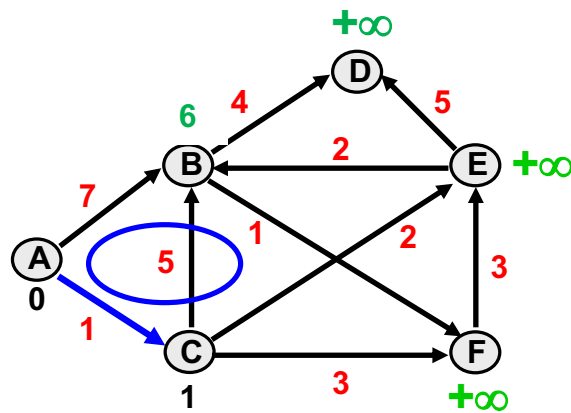
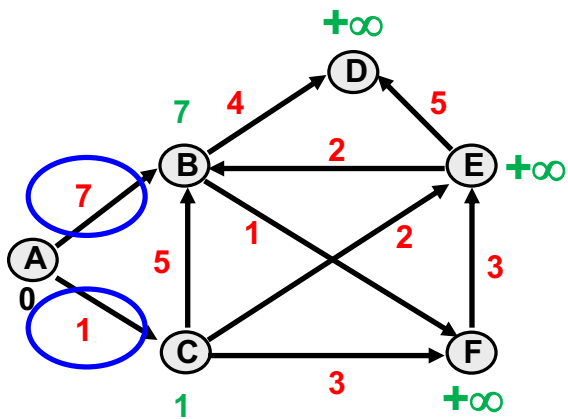
Os arcos indicados são os últimos que atualizaram a distância



Exemplo do algoritmo de Dijkstra

Pesos dos arcos, distâncias provisórias, distâncias definitivas

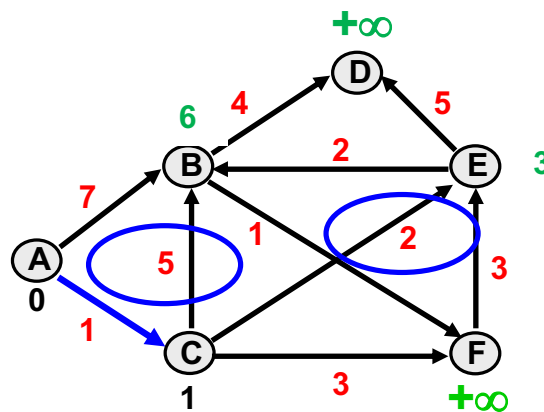
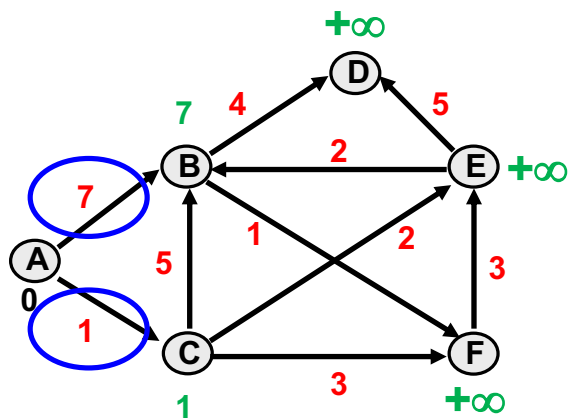
Os arcos indicados são os últimos que atualizaram a distância



Exemplo do algoritmo de Dijkstra

Pesos dos arcos, distâncias provisórias, distâncias definitivas

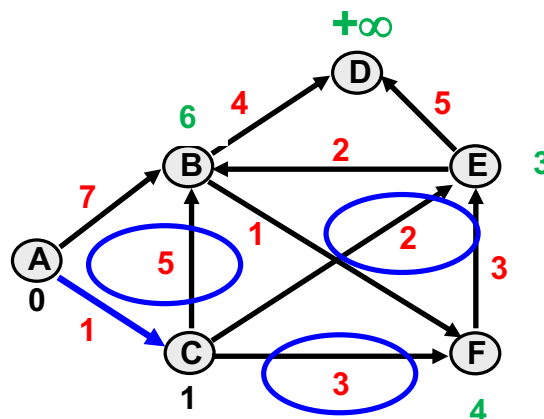
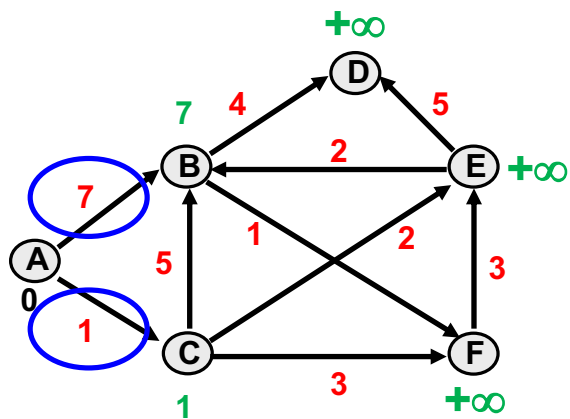
Os arcos indicados são os últimos que atualizaram a distância



Exemplo do algoritmo de Dijkstra

Pesos dos arcos, distâncias provisórias, distâncias definitivas

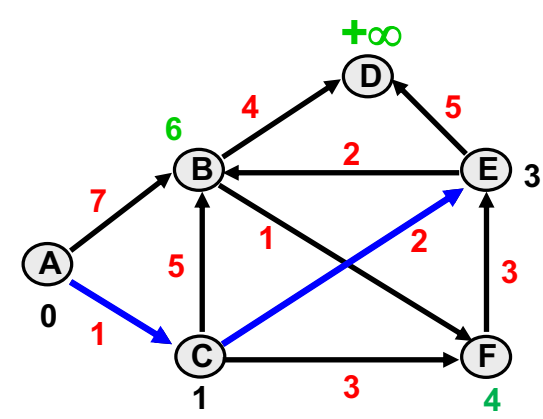
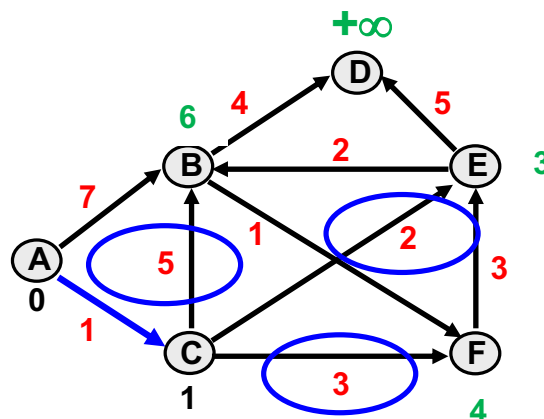
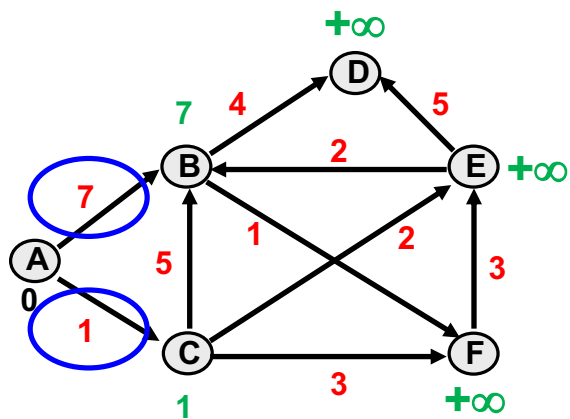
Os arcos indicados são os últimos que atualizaram a distância



Exemplo do algoritmo de Dijkstra

Pesos dos arcos, distâncias provisórias, distâncias definitivas

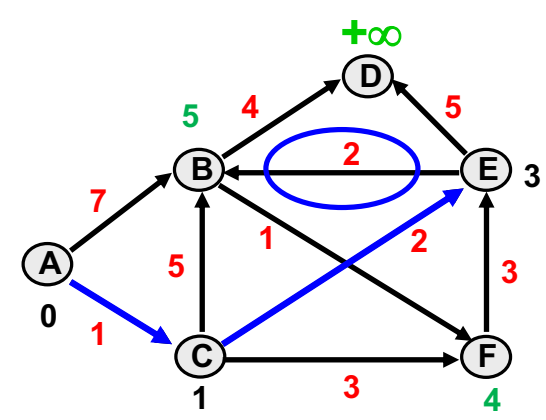
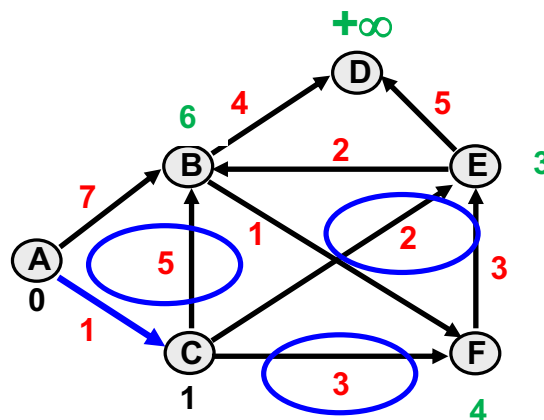
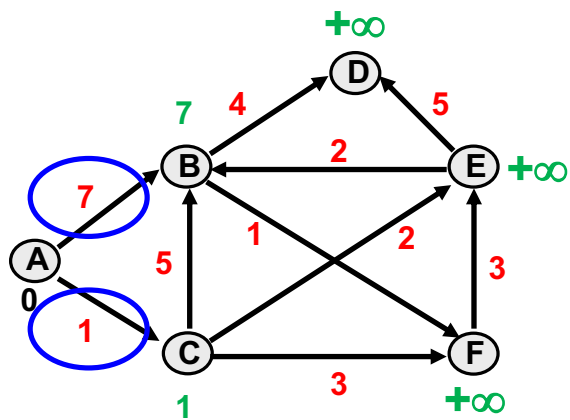
Os arcos indicados são os últimos que atualizaram a distância



Exemplo do algoritmo de Dijkstra

Pesos dos arcos, distâncias provisórias, distâncias definitivas

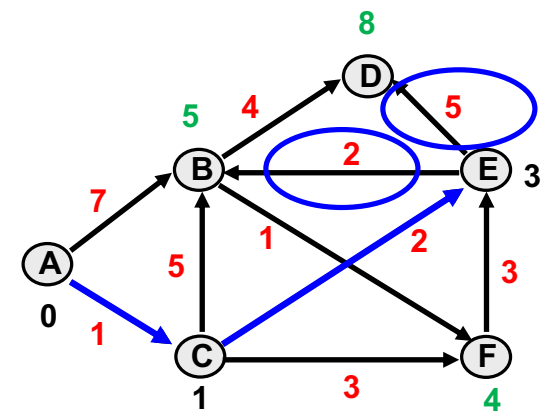
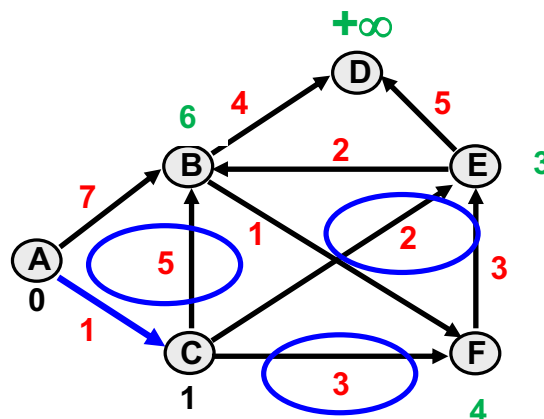
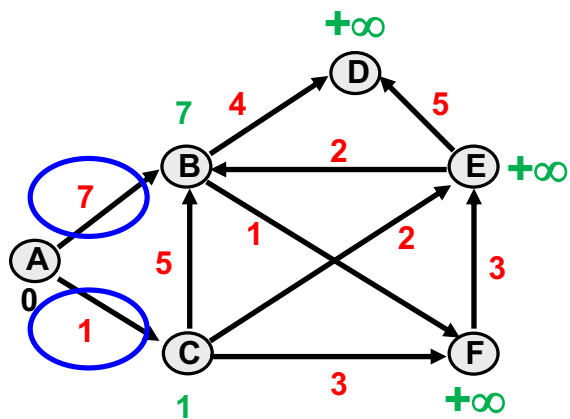
Os arcos indicados são os últimos que atualizaram a distância



Exemplo do algoritmo de Dijkstra

Pesos dos arcos, distâncias provisórias, distâncias definitivas

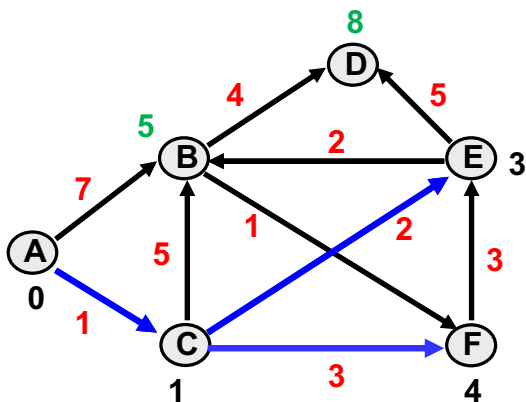
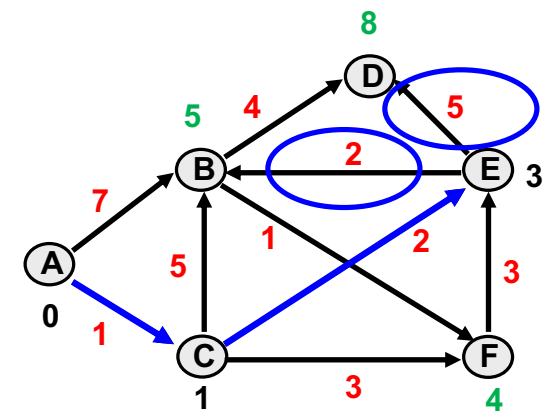
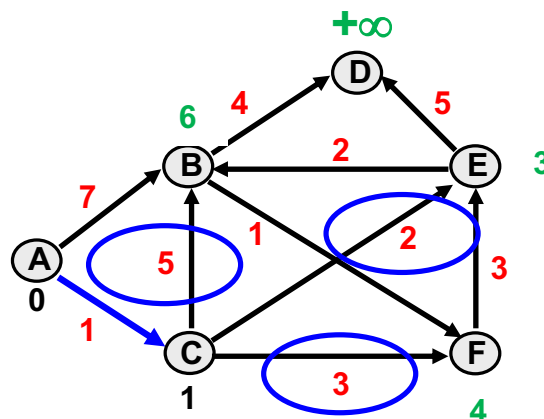
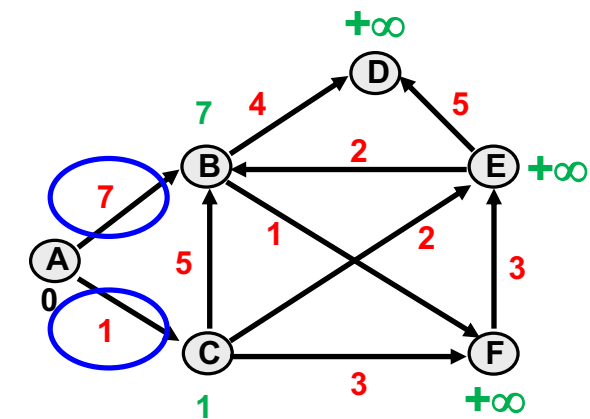
Os arcos indicados são os últimos que atualizaram a distância



Exemplo do algoritmo de Dijkstra

Pesos dos arcos, distâncias provisórias, distâncias definitivas

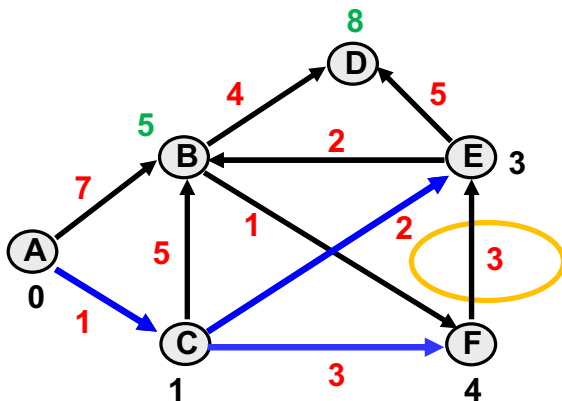
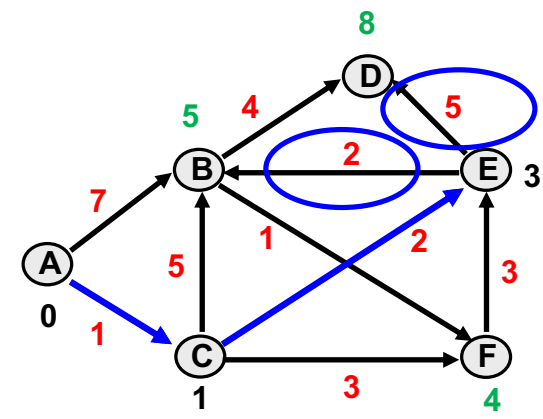
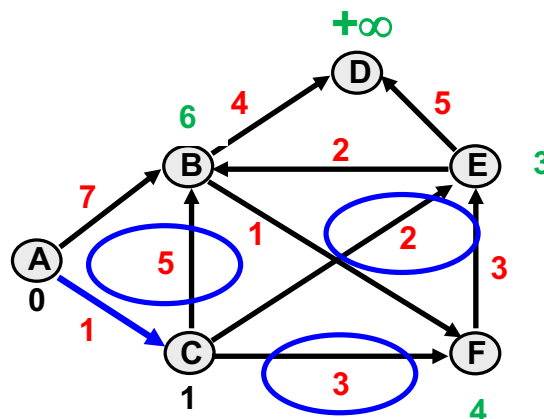
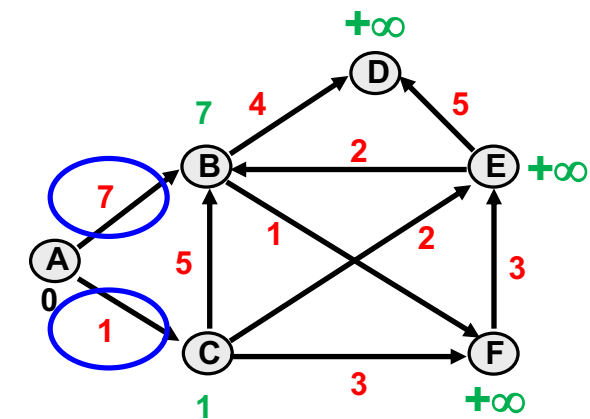
Os arcos indicados são os últimos que atualizaram a distância



Exemplo do algoritmo de Dijkstra

Pesos dos arcos, distâncias provisórias, distâncias definitivas

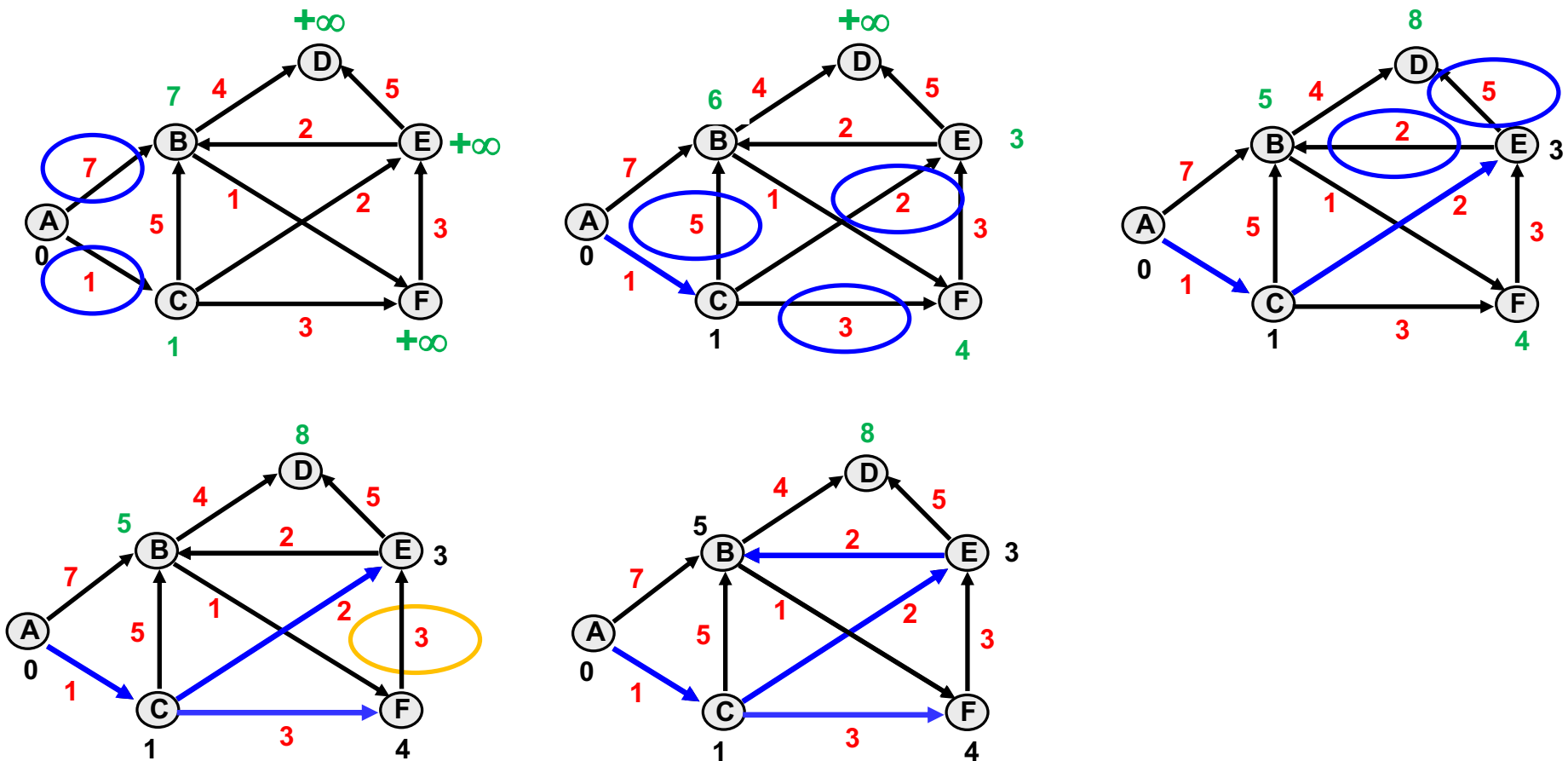
Os arcos indicados são os últimos que atualizaram a distância



Exemplo do algoritmo de Dijkstra

Pesos dos arcos, distâncias provisórias, distâncias definitivas

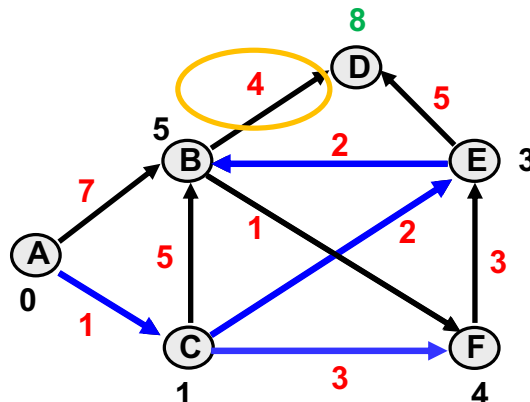
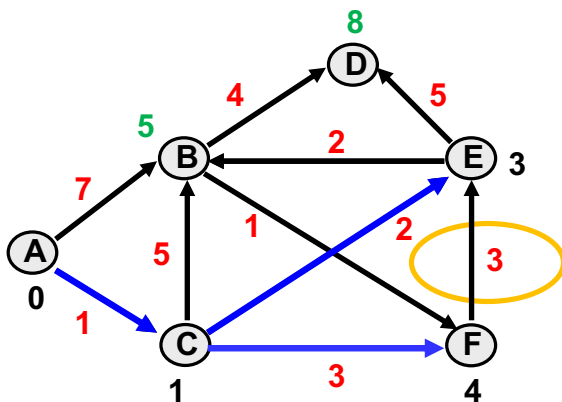
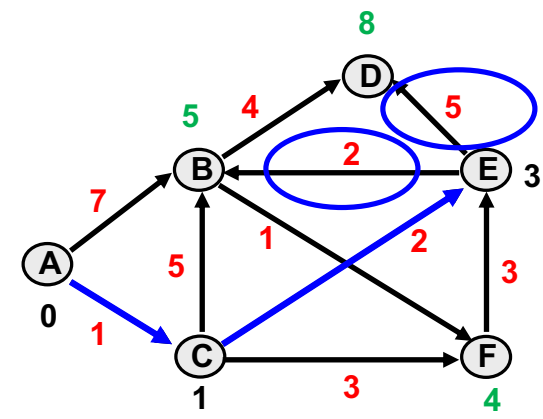
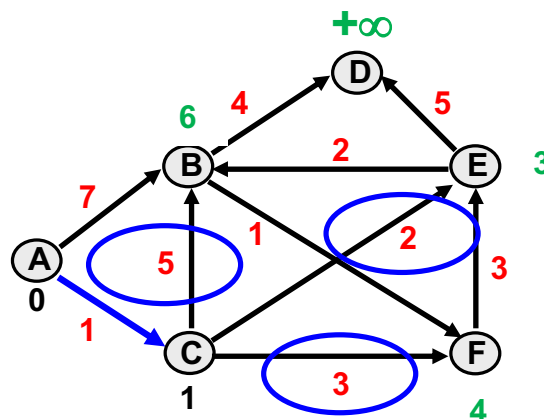
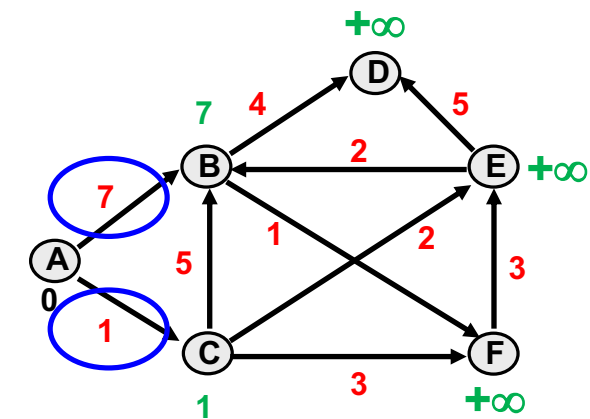
Os arcos indicados são os últimos que atualizaram a distância



Exemplo do algoritmo de Dijkstra

Pesos dos arcos, distâncias provisórias, distâncias definitivas

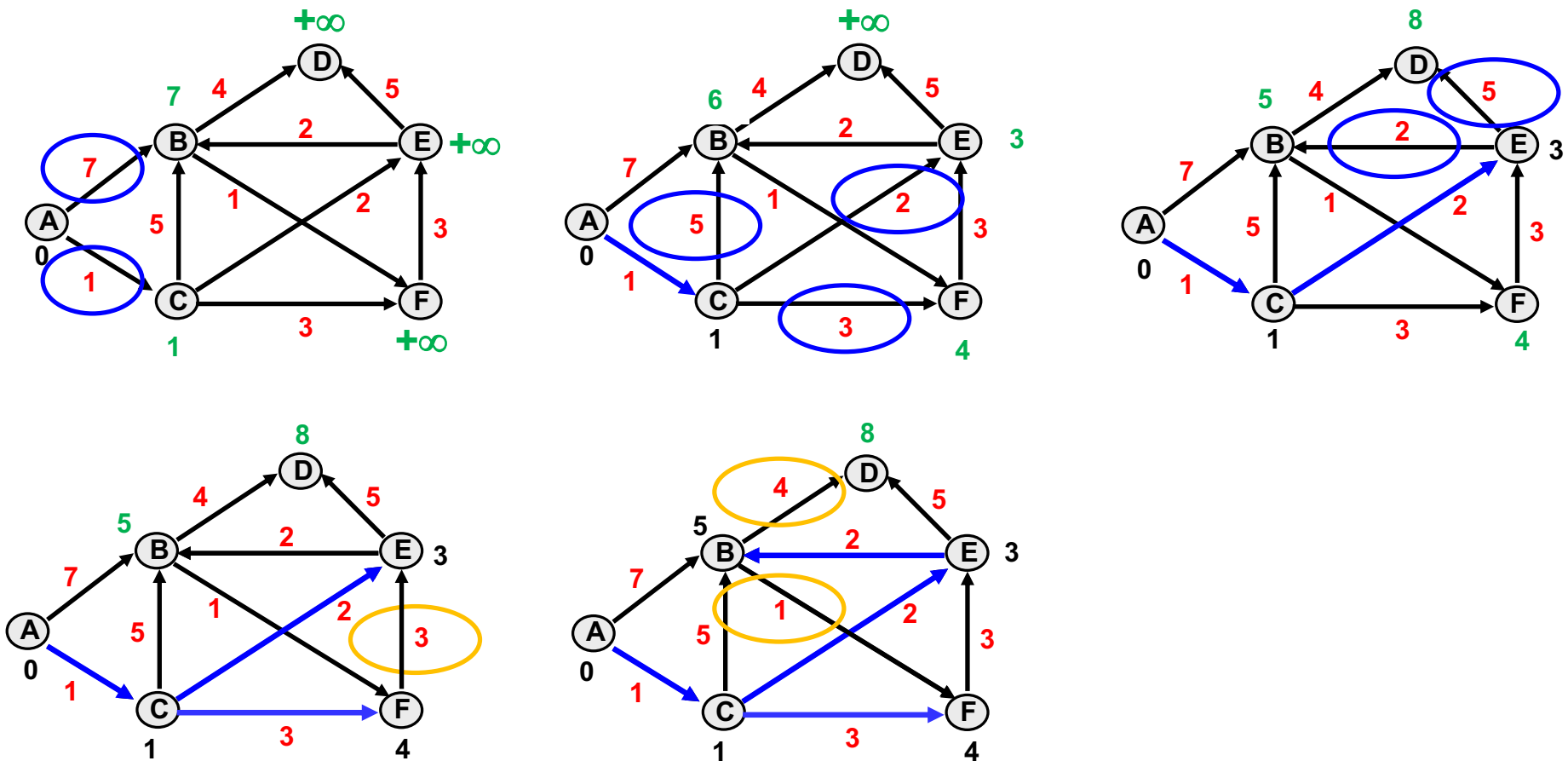
Os arcos indicados são os últimos que atualizaram a distância



Exemplo do algoritmo de Dijkstra

Pesos dos arcos, distâncias provisórias, distâncias definitivas

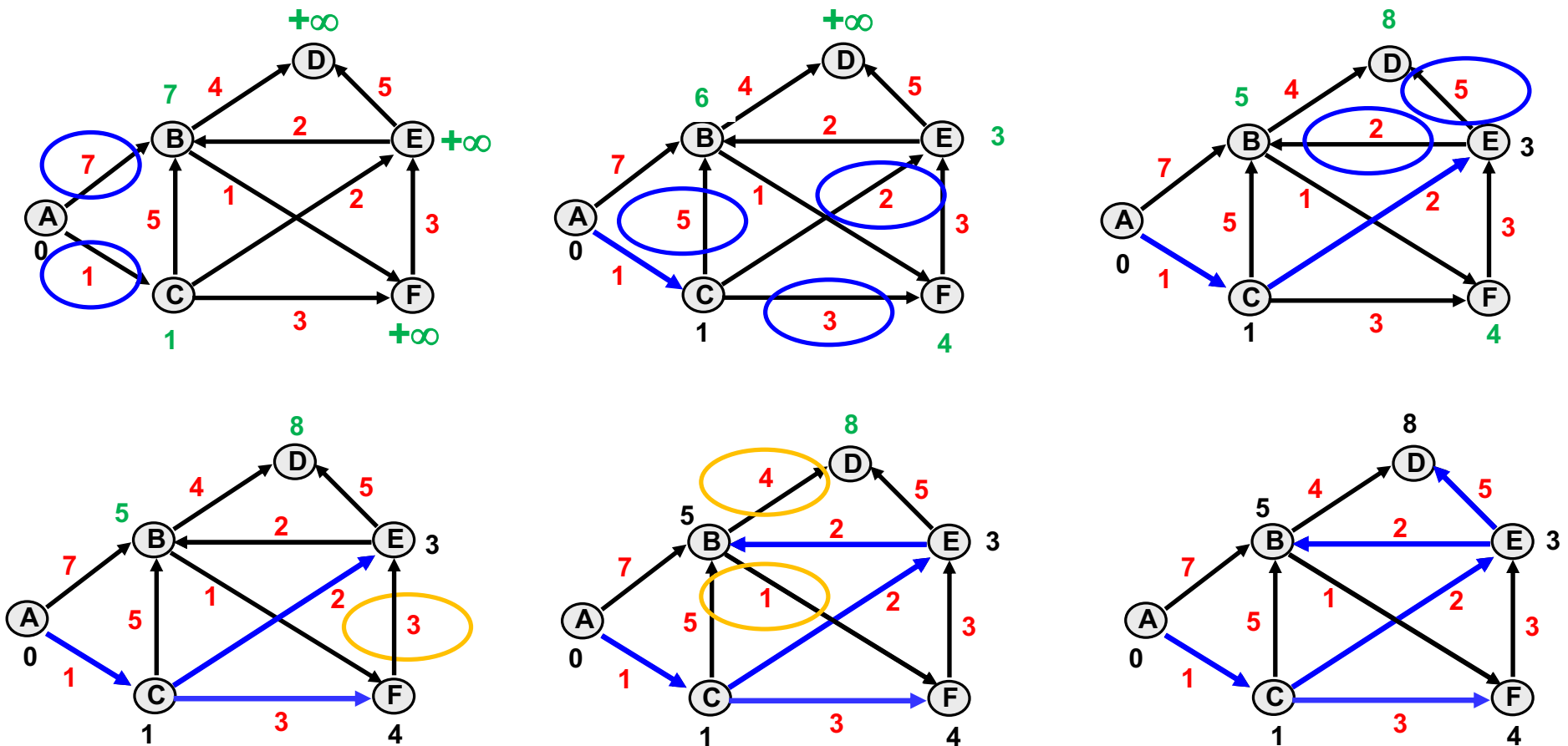
Os arcos indicados são os últimos que atualizaram a distância



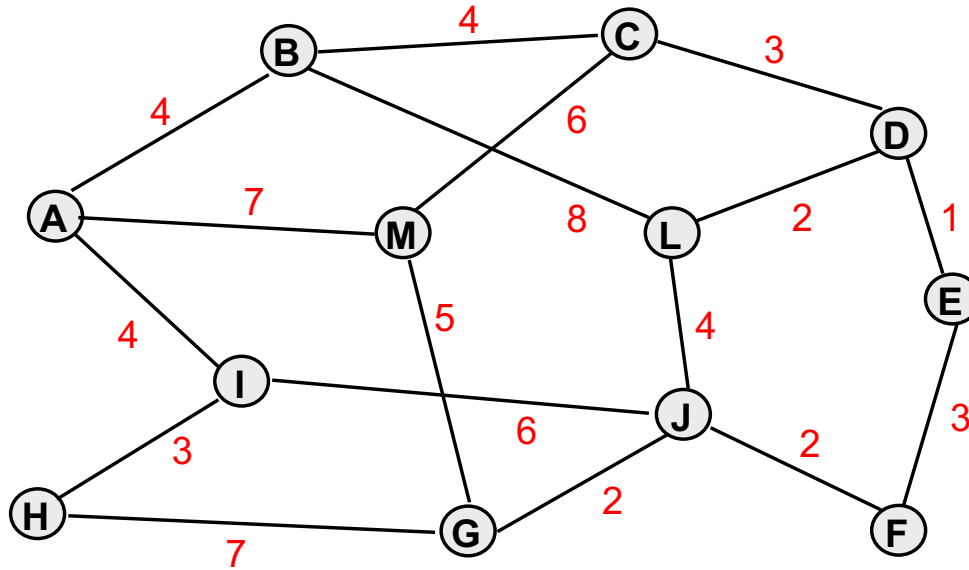
Exemplo do algoritmo de Dijkstra

Pesos dos arcos, distâncias provisórias, distâncias definitivas

Os arcos indicados são os últimos que atualizaram a distância



Exemplo



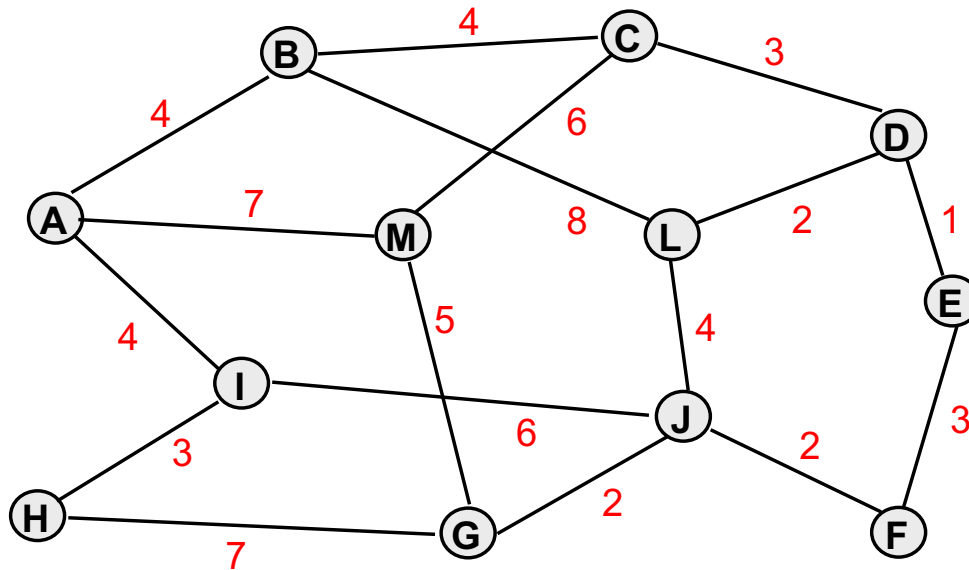
Componentes

{A}	{B}	{C}	{D}	{E}	{F}
{G}	{H}	{I}	{J}	{L}	{M}

Lista ordenada

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



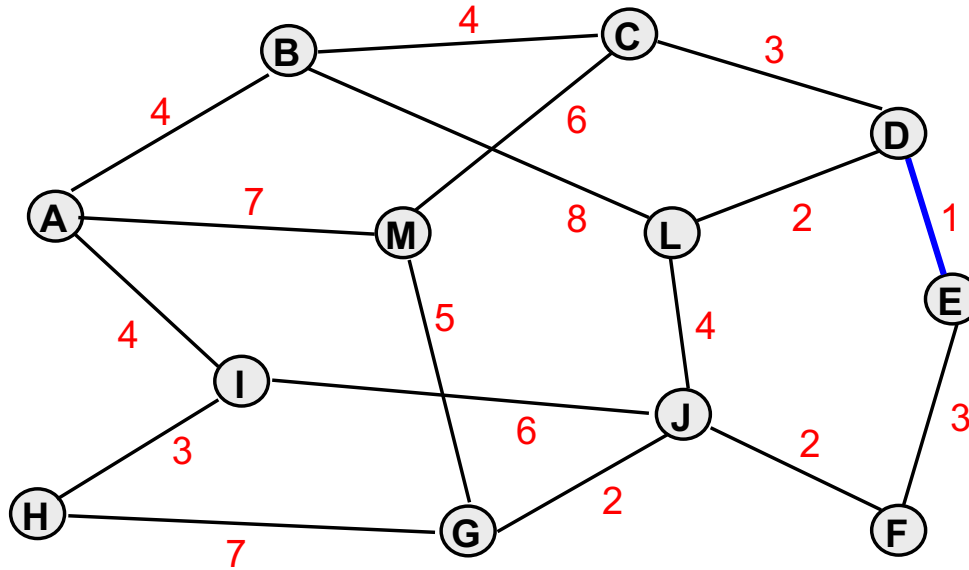
Componentes

{A}	{B}	{C}	{D}	{E}	{F}
{G}	{H}	{I}	{J}	{L}	{M}

Lista ordenada

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



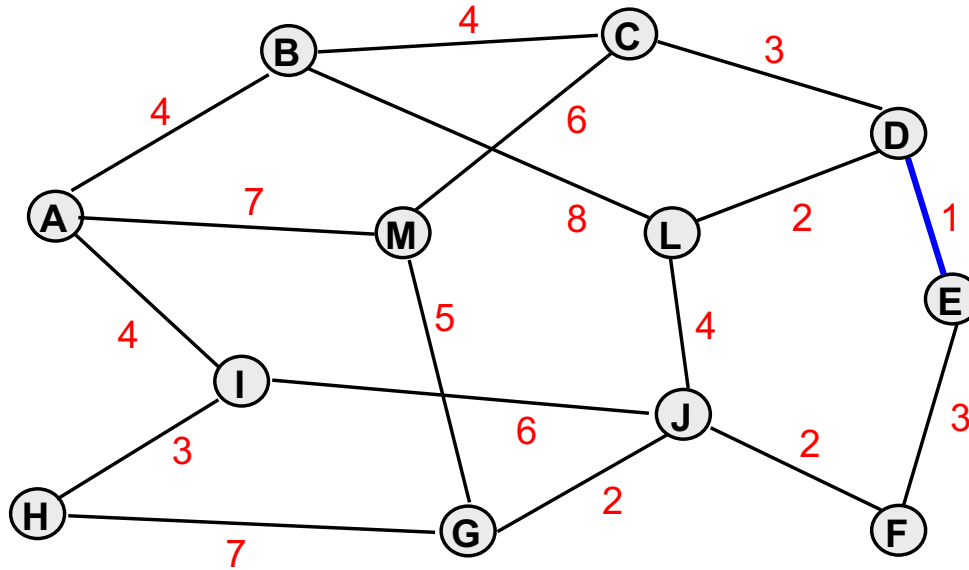
Componentes

{A}	{B}	{C}	{D}	{E}	{F}
{G}	{H}	{I}	{J}	{L}	{M}

Lista ordenada

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



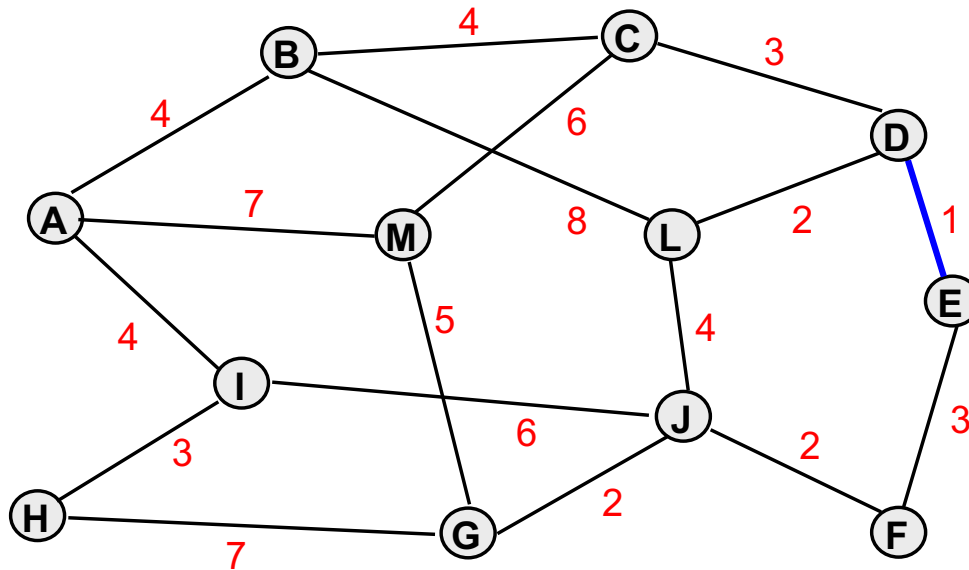
Componentes

{A} {B} {C} {D,E} {F}
{G} {H} {I} {J} {L} {M}

Lista ordenada

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 1$$

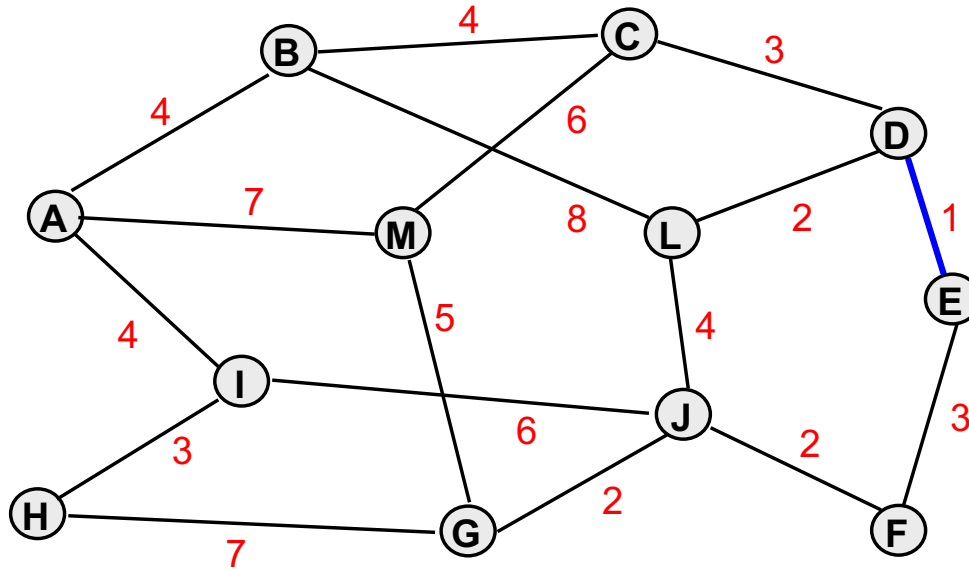
Componentes

{A}	{B}	{C}	{D,E}	{F}	
{G}	{H}	{I}	{J}	{L}	{M}

**Lista
ordenada**

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 1$$

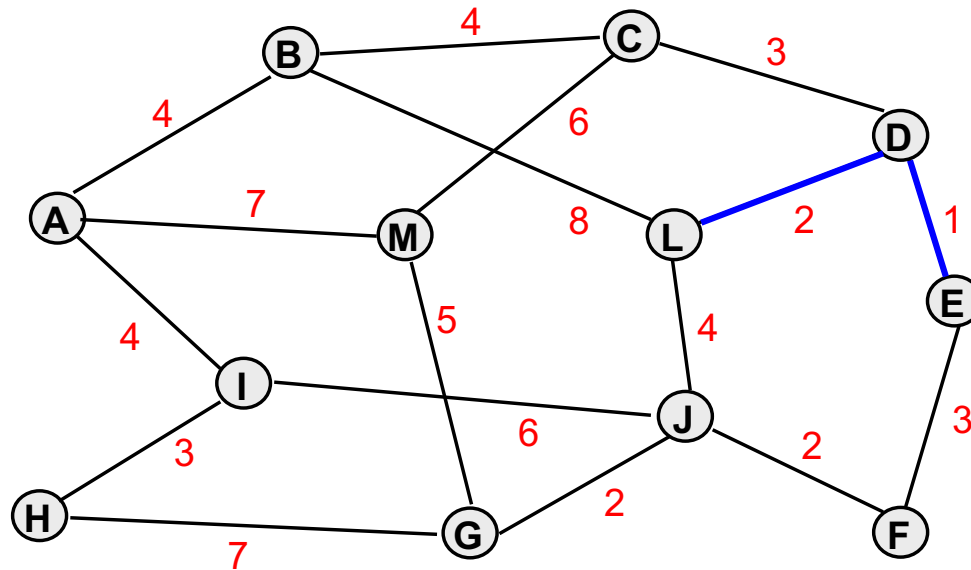
Componentes

{A} {B} {C} {D,E} {F}
{G} {H} {I} {J} {L} {M}

**Lista
ordenada**

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 1$$

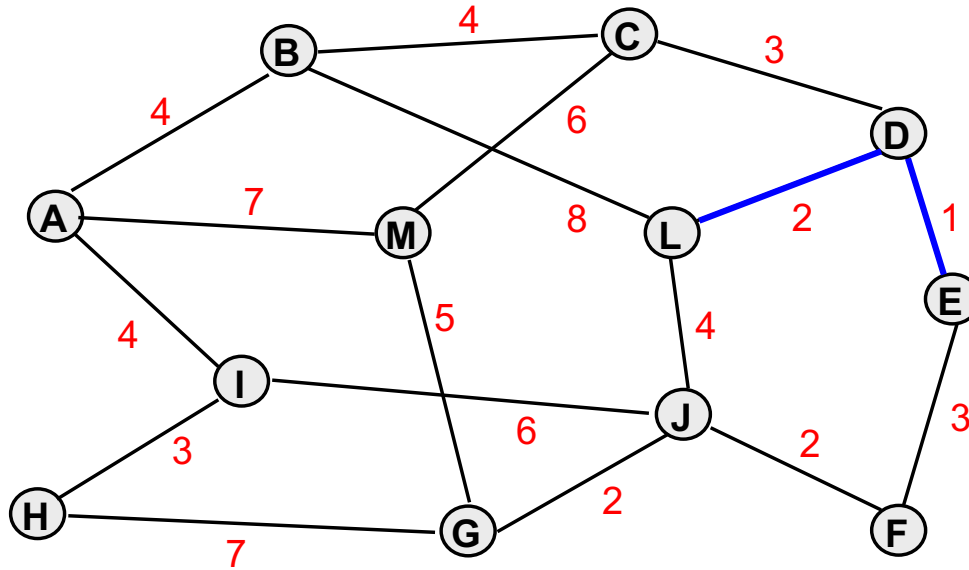
Componentes

{A} {B} {C} {D,E} {F}
{G} {H} {I} {J} {L} {M}

**Lista
ordenada**

<i>e</i>	<i>c(e)</i>
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 1$$

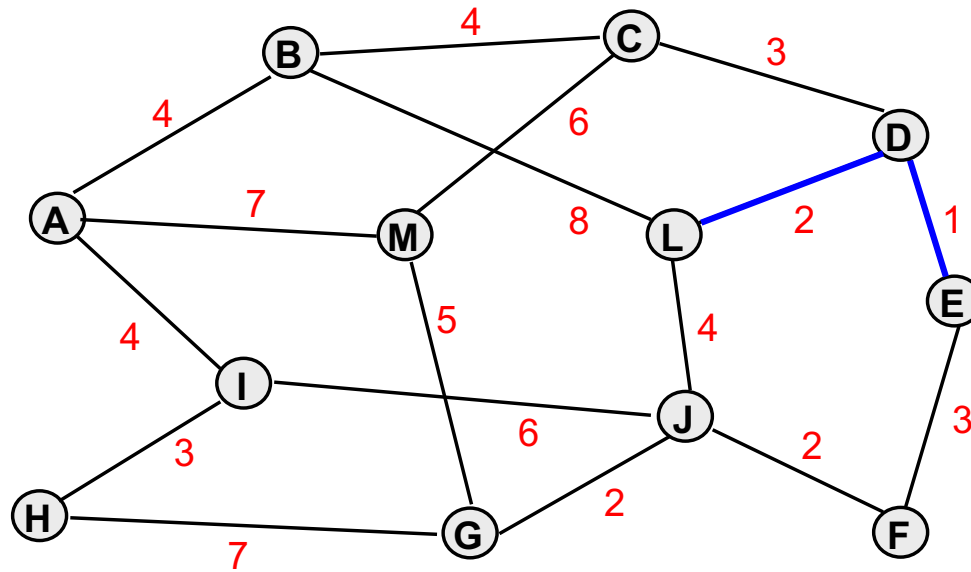
Componentes

{A} {B} {C} {D, E, L} {F}
{G} {H} {I} {J} {M}

**Lista
ordenada**

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 3$$

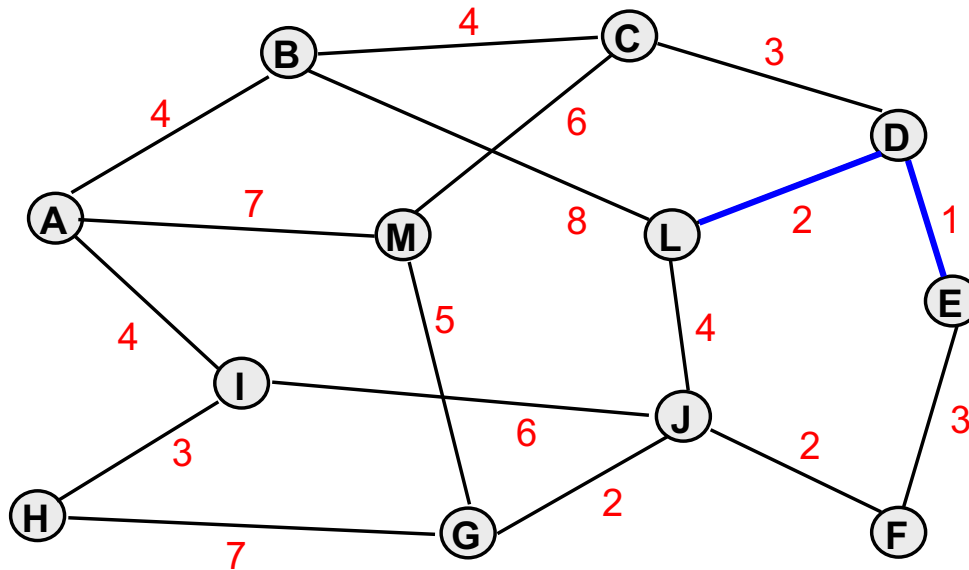
Componentes

{A} {B} {C} {D, E, L} {F}
{G} {H} {I} {J} {M}

**Lista
ordenada**

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 3$$

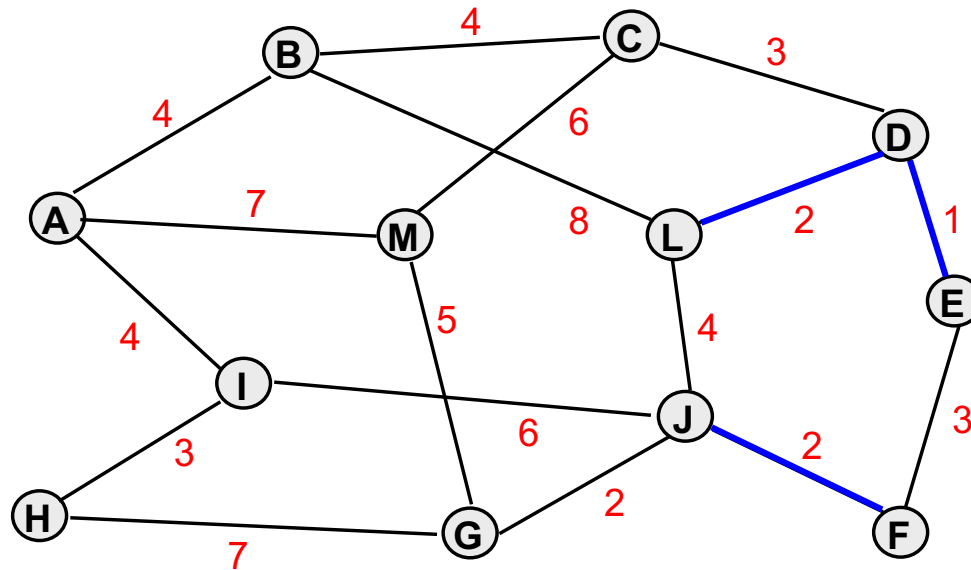
Componentes

{A} {B} {C} {D, E, L} {F}
{G} {H} {I} {J} {M}

**Lista
ordenada**

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$c(T) = 3$

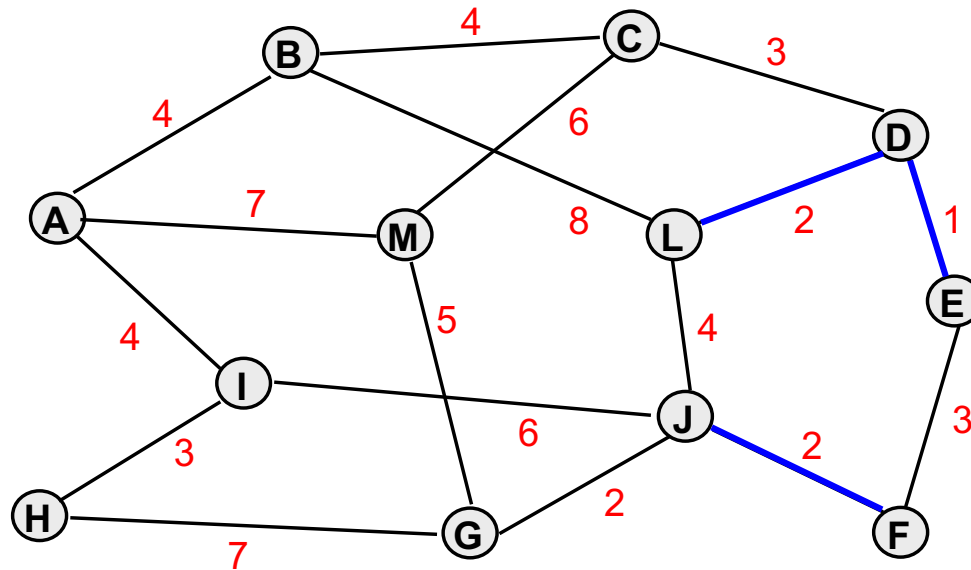
Componentes

{A}	{B}	{C}	{D, E, L}	{F}
{G}	{H}	{I}	{J}	{M}

Lista ordenada

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$c(T) = 3$

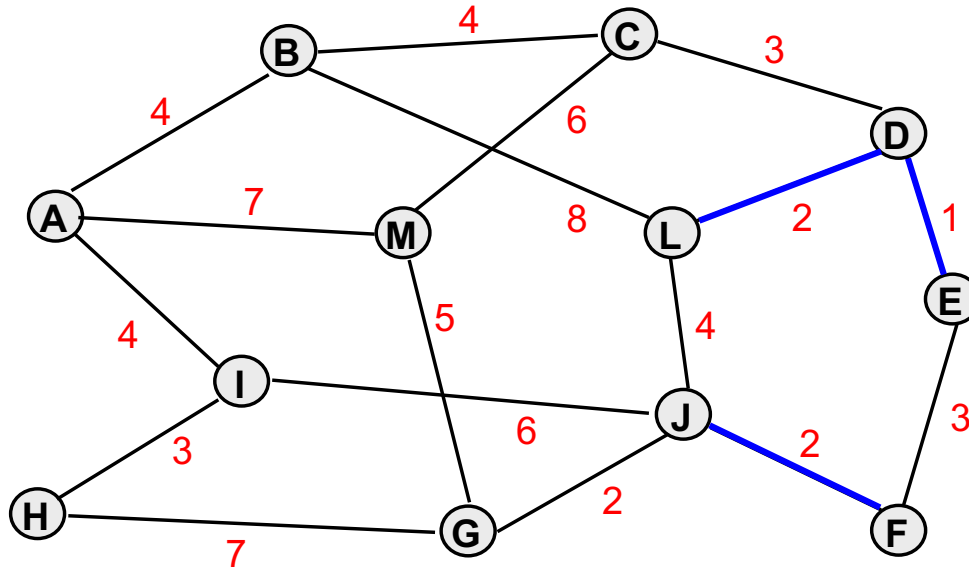
Componentes

{A}	{B}	{C}	{D, E, L}	
{F, J}	{G}	{H}	{I}	{M}

Lista ordenada

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 5$$

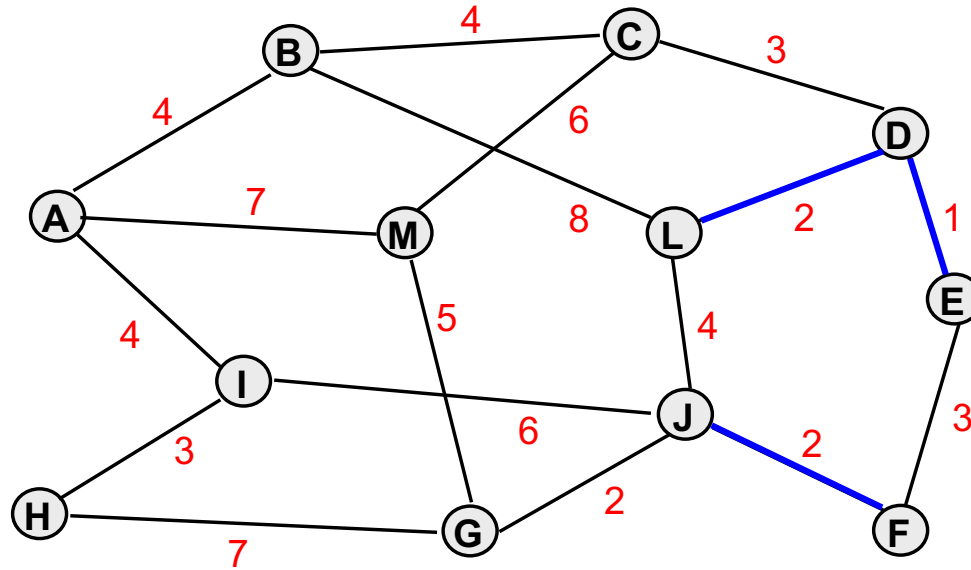
Componentes

{A}	{B}	{C}	{D, E, L}	
{F, J}	{G}	{H}	{I}	{M}

Lista ordenada

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$c(T) = 5$

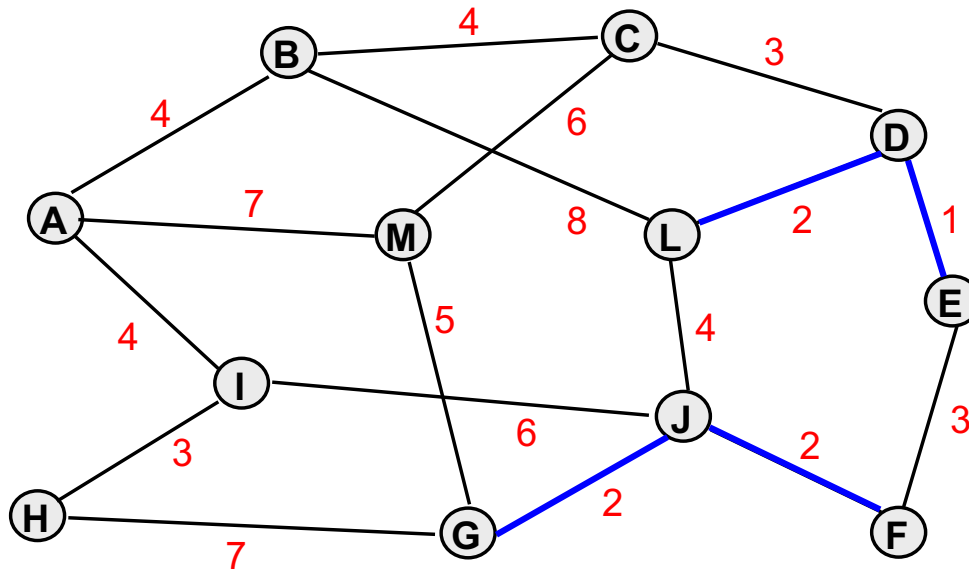
Componentes

{A}	{B}	{C}	{D, E, L}	
{F, J}	{G}	{H}	{I}	{M}

Lista ordenada

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$c(T) = 5$

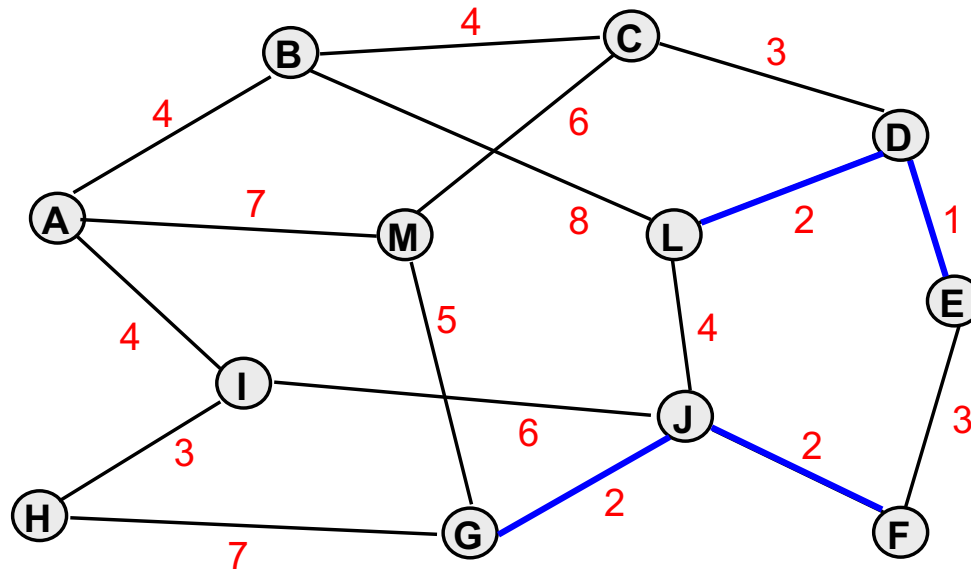
Componentes

{A}	{B}	{C}	{D, E, L}	
{F, J}	{G}	{H}	{I}	{M}

Lista ordenada

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 5$$

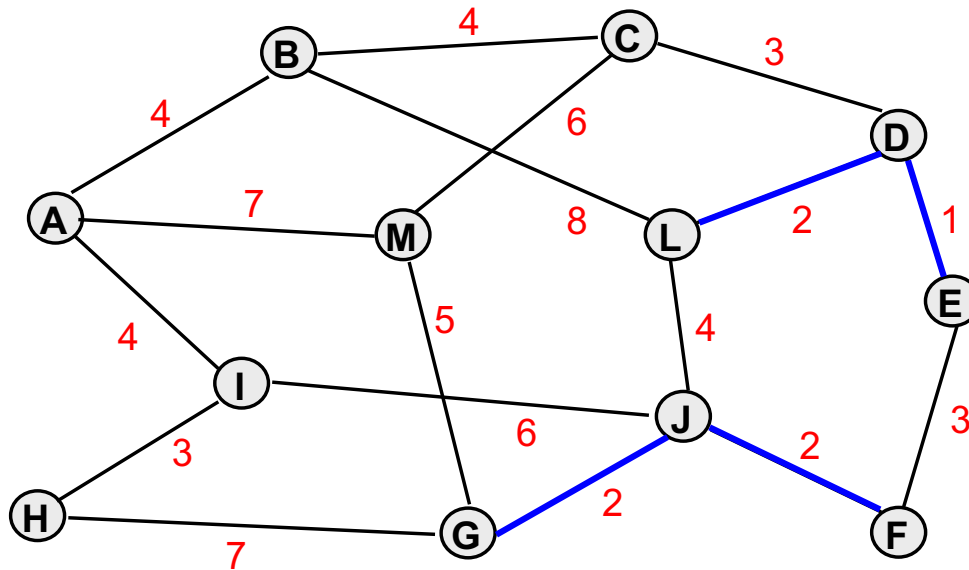
Componentes

{A} {B} {C} {D, E, L}
{F, G, J} {H} {I} {M}

**Lista
ordenada**

<i>e</i>	<i>c(e)</i>
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 7$$

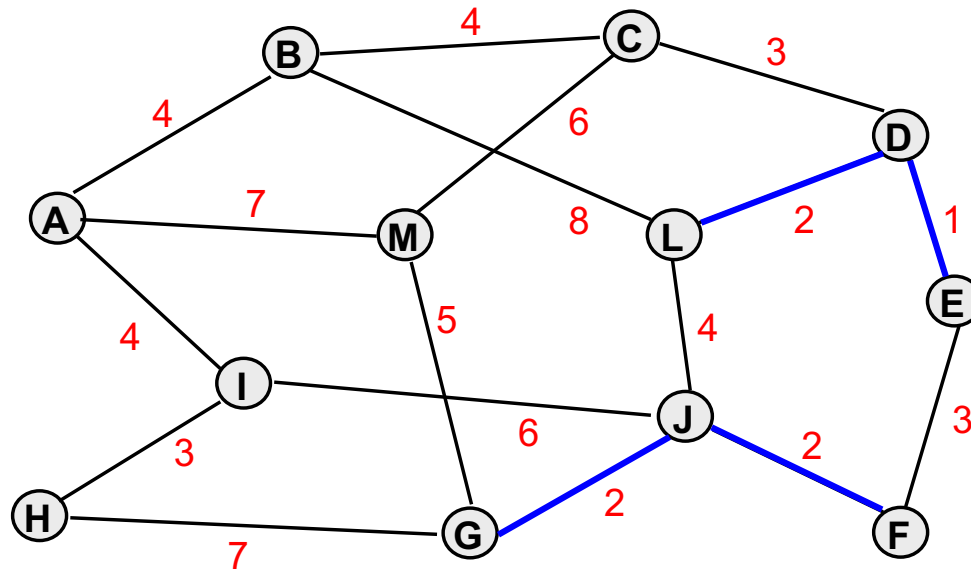
Componentes

{A} {B} {C} {D, E, L}
{F, G, J} {H} {I} {M}

**Lista
ordenada**

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 7$$

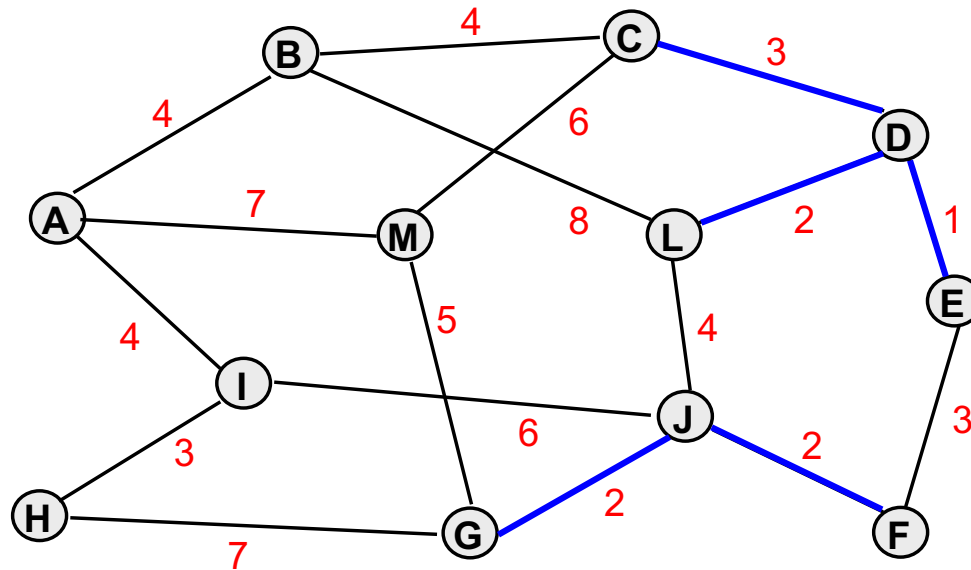
Componentes

{A} {B} {C} {D, E, L}
{F, G, J} {H} {I} {M}

**Lista
ordenada**

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$c(T) = 7$

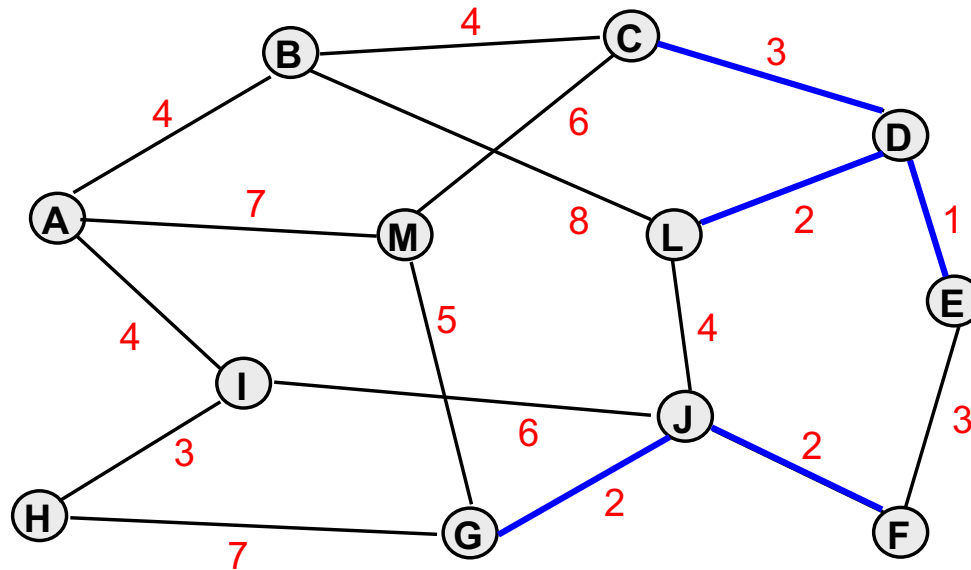
Componentes

{A}	{B}	{C}	{D, E, L}
{F, G, J}	{H}	{I}	{M}

Lista ordenada

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 7$$

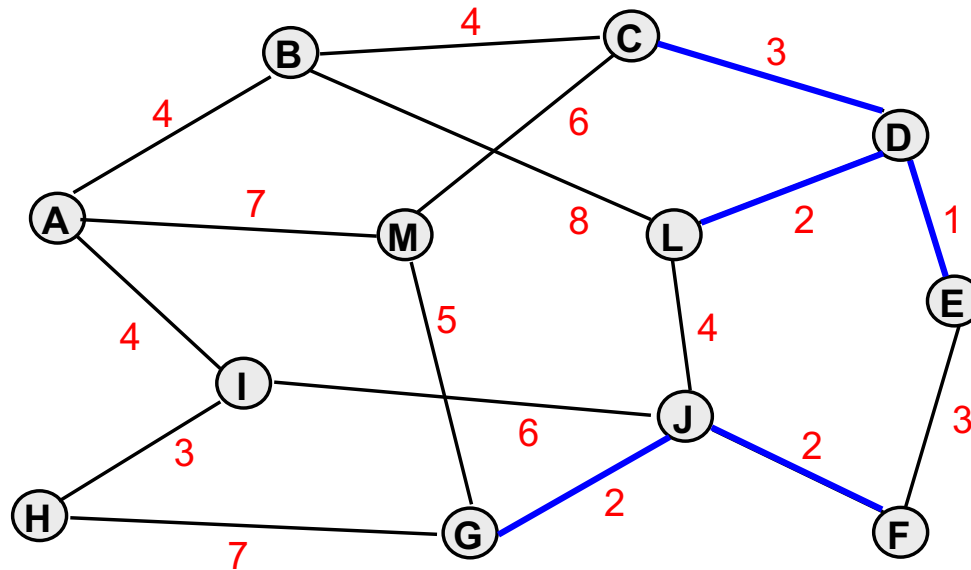
Componentes

{A}	{B}	{C, D, E, L}	
{F, G, J}	{H}	{I}	{M}

Lista ordenada

<i>e</i>	<i>c(e)</i>
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$c(T) = 10$

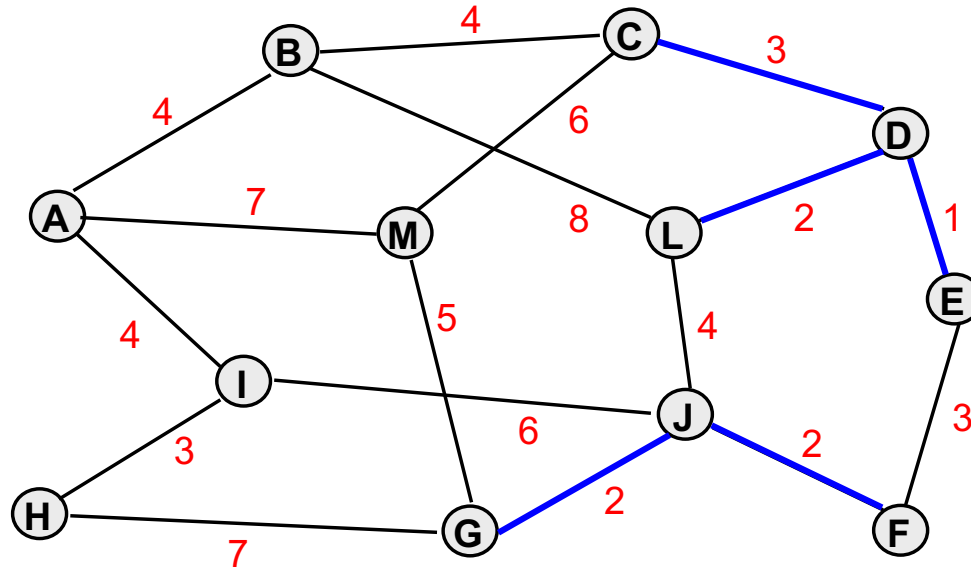
Componentes

{A}	{B}	{C, D, E, L}
{F, G, J}	{H}	{I} {M}

Lista ordenada

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 10$$

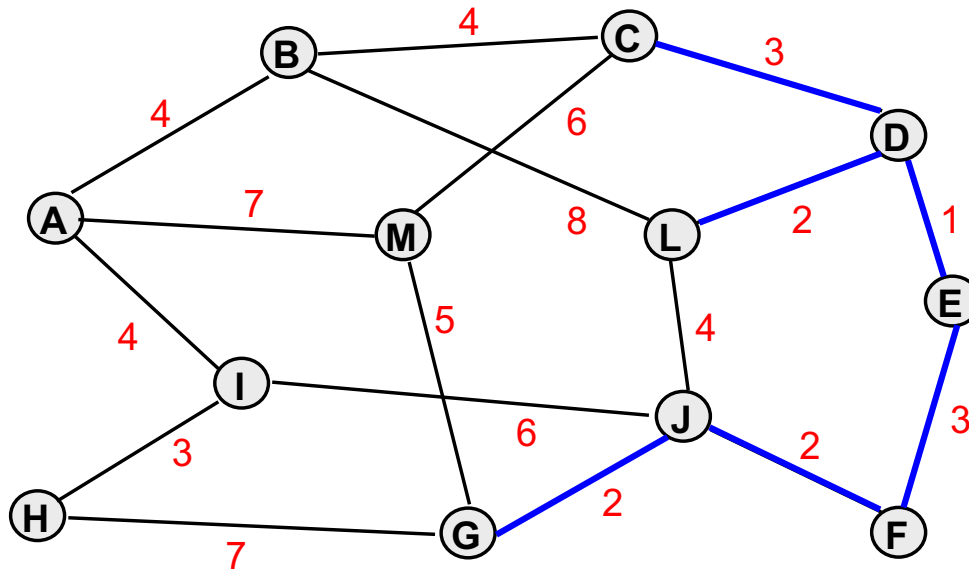
Componentes

{A}	{B}	{C, D, E, L}	
{F, G, J}	{H}	{I}	{M}

Lista ordenada

<i>e</i>	<i>c(e)</i>
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 10$$

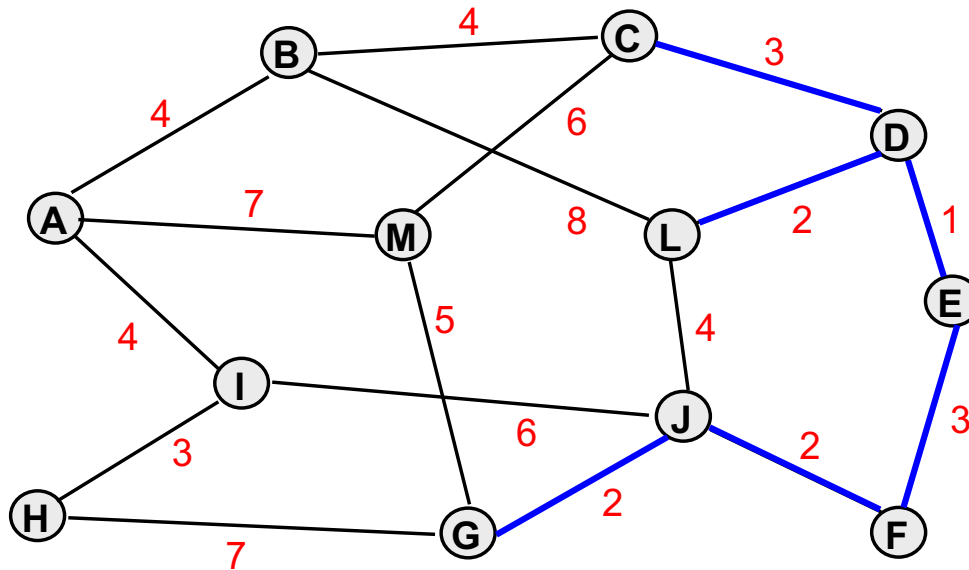
Componentes

{A}	{B}	{C, D, E, L}	
{F, G, J}	{H}	{I}	{M}

Lista ordenada

<i>e</i>	<i>c(e)</i>
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$c(T) = 10$

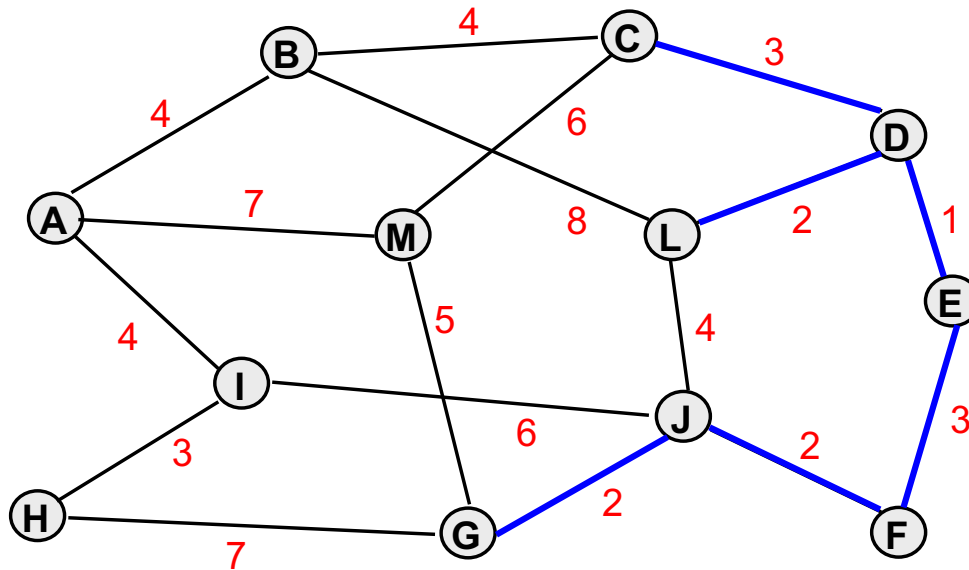
Componentes

{A}	{B}	{C, D, E, L, F, G, J}
	{H}	{I}
		{M}

Lista ordenada

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$c(T) = 13$

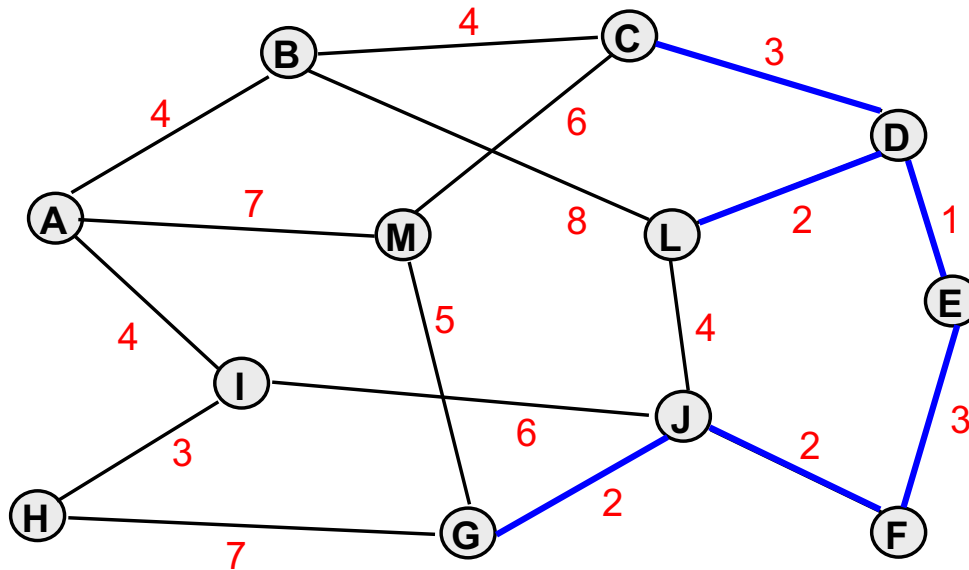
Componentes

{A}	{B}	{C, D, E, L, F, G, J}
	{H}	{I}
		{M}

Lista ordenada

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 13$$

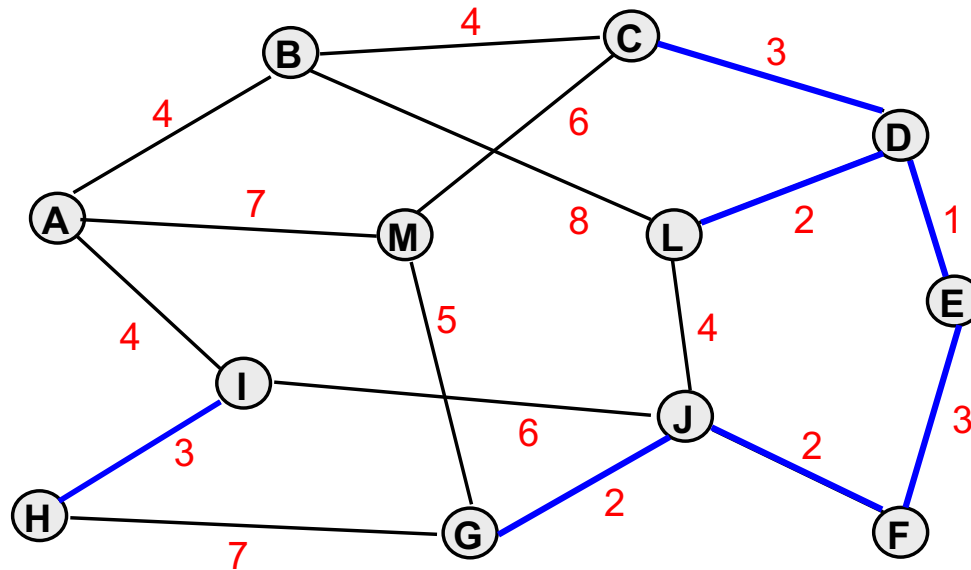
Componentes

{A} {B} {C, D, E, L, F, G, J}
{H} {I} {M}

**Lista
ordenada**

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 13$$

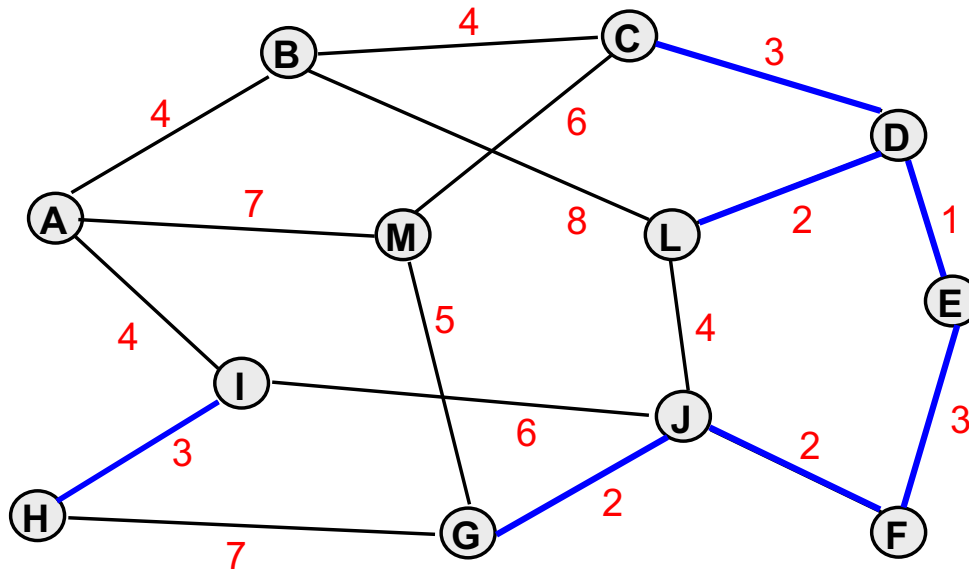
Componentes

{A} {B} {C, D, E, L, F, G, J}
{H} {I} {M}

**Lista
ordenada**

<i>e</i>	<i>c(e)</i>
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$c(T) = 13$

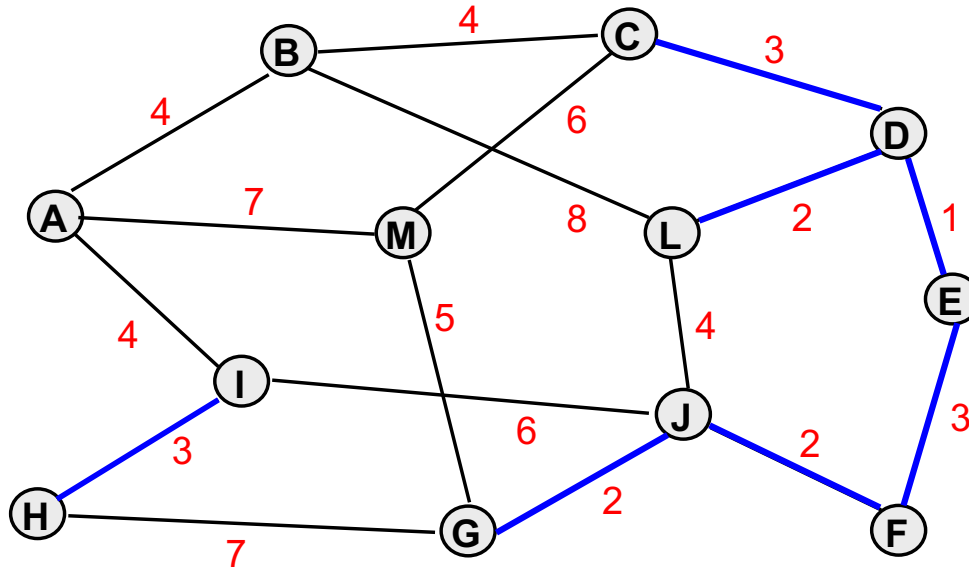
Componentes

{A}	{B}	{C, D, E, L, F, G, J}
	{H, I}	{M}

Lista ordenada

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 16$$

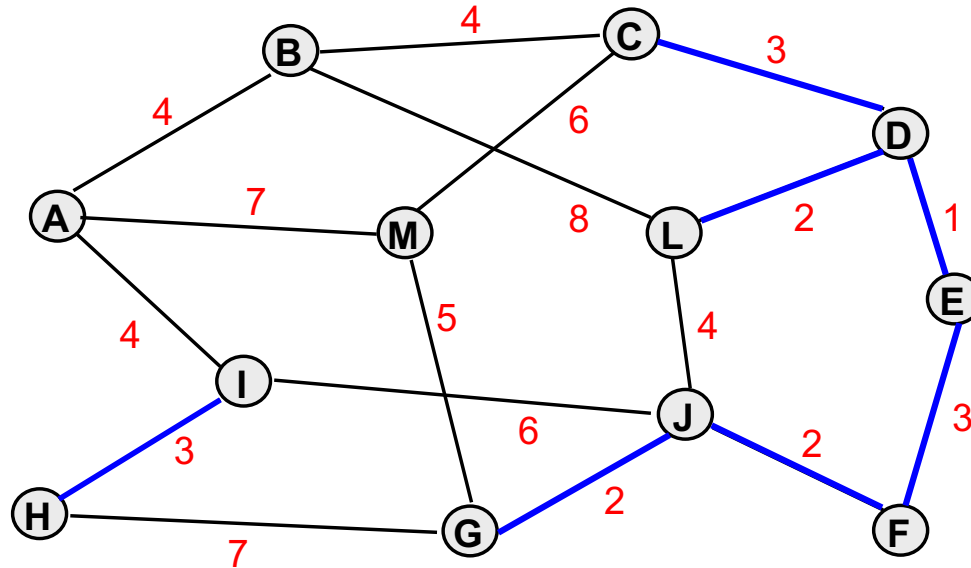
Componentes

{A} {B} {C, D, E, L, F, G, J}
{H, I} {M}

**Lista
ordenada**

e	c(e)
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$c(T) = 16$

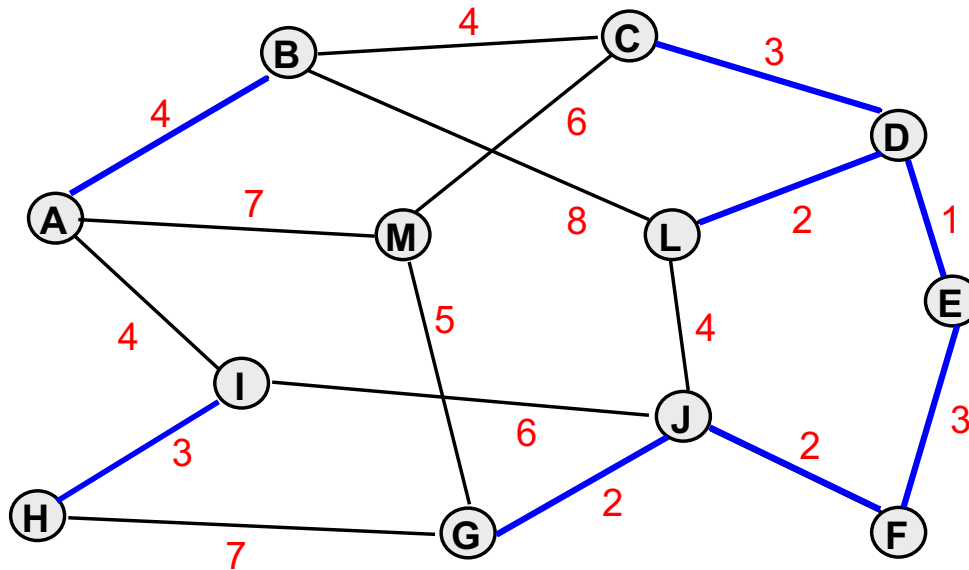
Componentes

{A}	{B}	{C, D, E, L, F, G, J}
	{H, I}	{M}

Lista ordenada

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$c(T) = 16$

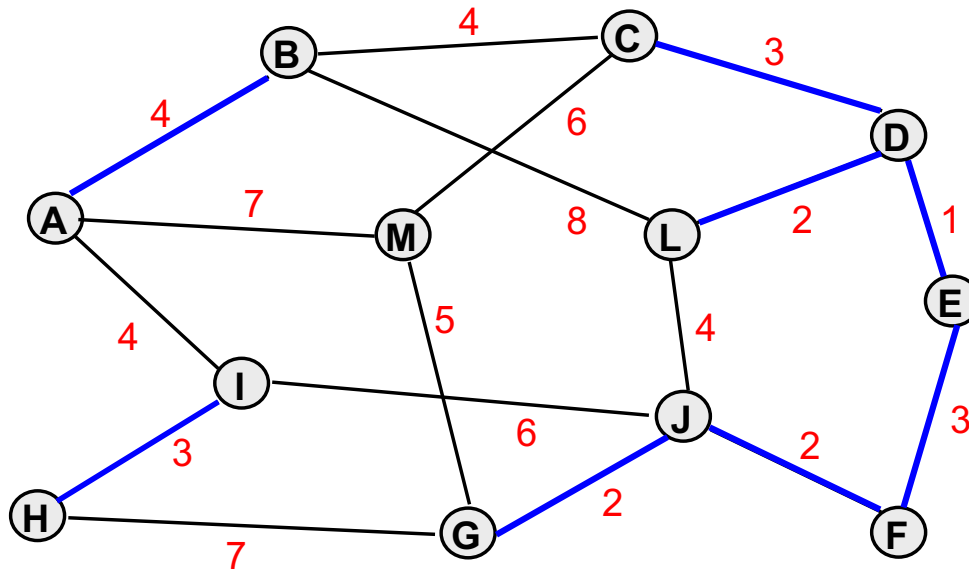
Componentes

{A}	{B}	{C, D, E, L, F, G, J}
	{H, I}	{M}

Lista ordenada

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 16$$

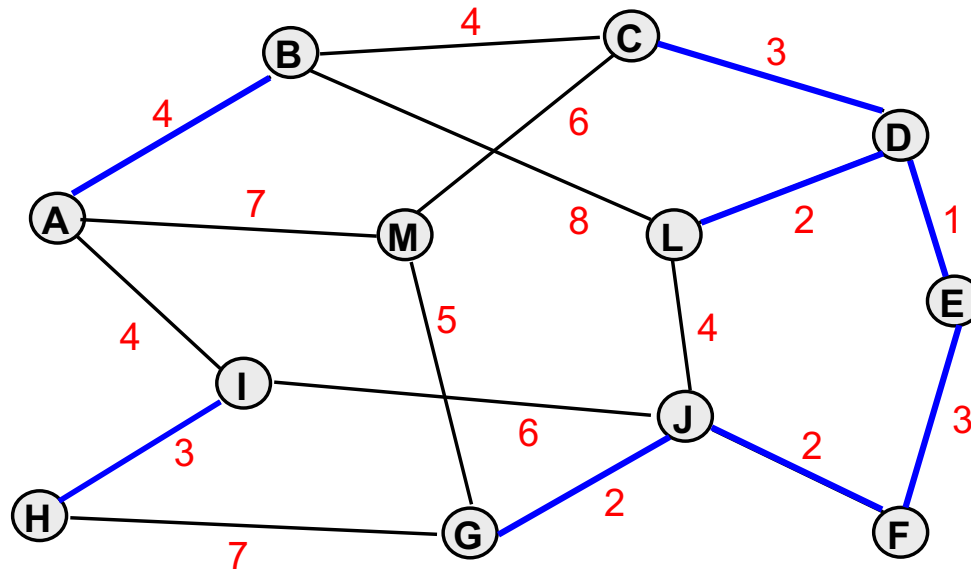
Componentes

$\{A, B\}$ $\{C, D, E, L, F, G, J\}$
 $\{H, I\}$ $\{M\}$

**Lista
ordenada**

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 20$$

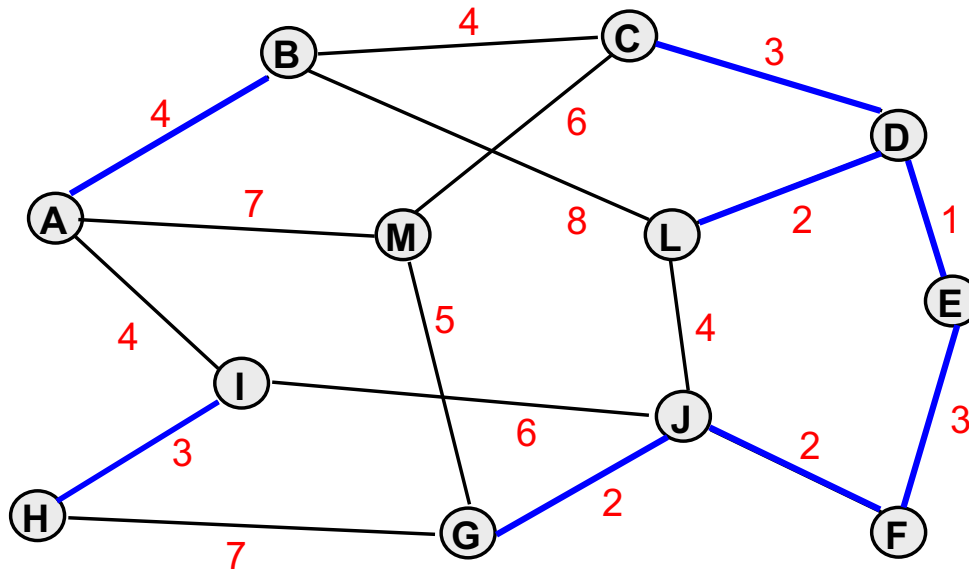
Componentes

$\{A, B\}$ $\{C, D, E, L, F, G, J\}$
 $\{H, I\}$ $\{M\}$

**Lista
ordenada**

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$c(T) = 20$

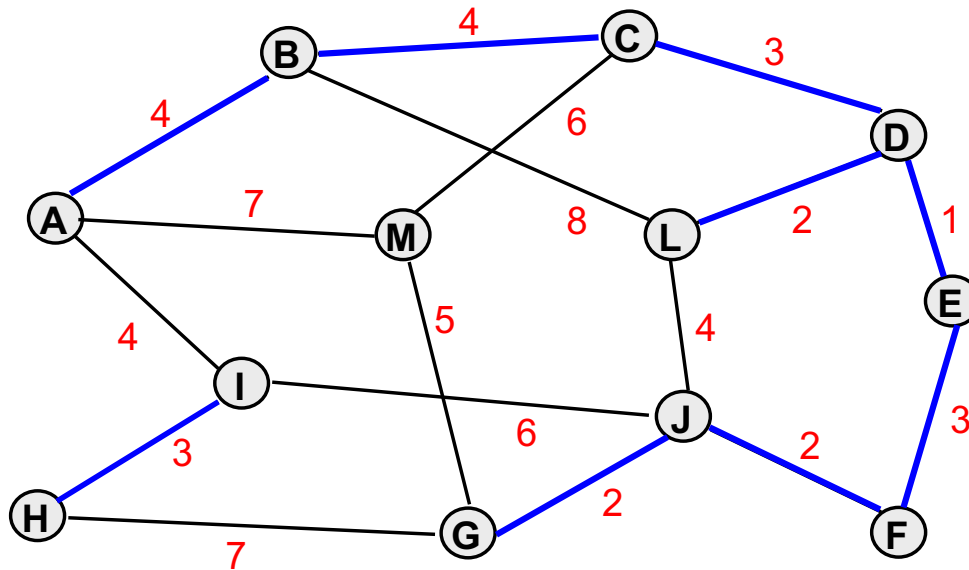
Componentes

{ A, B }	{ C, D, E, L, F, G, J }
{ H, I }	{ M }

Lista ordenada

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 20$$

Componentes

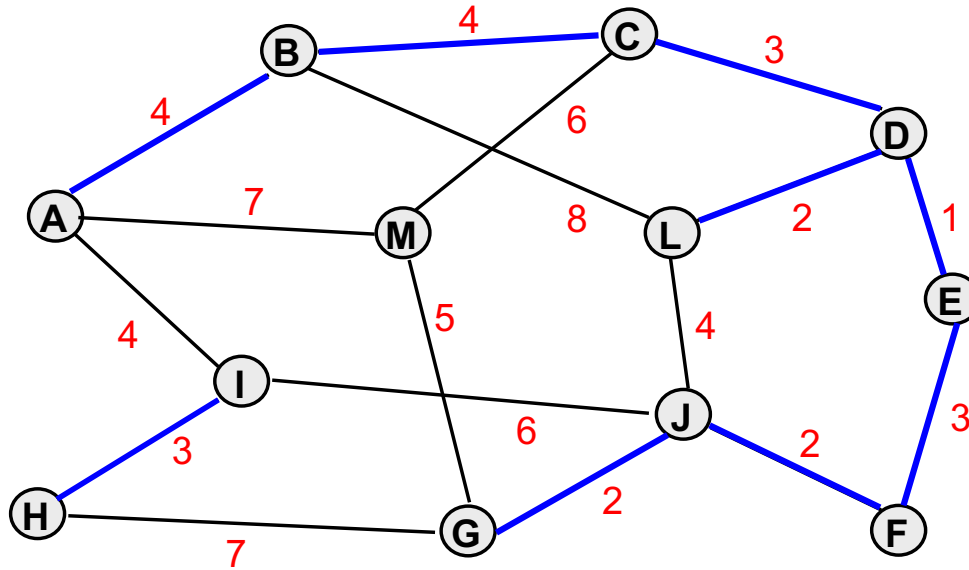
{ A, B, C, D, E, F, G, J, L }

{ H, I } { M }

**Lista
ordenada**

e	c(e)
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$c(T) = 24$

Componentes

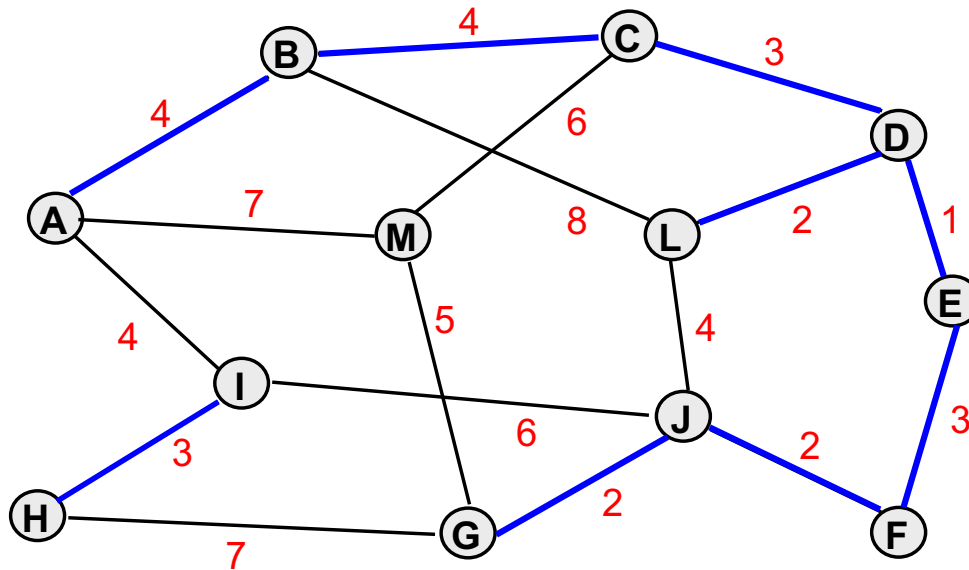
$\{A, B, C, D, E, F, G, J, L\}$

$\{H, I\} \quad \{M\}$

Lista ordenada

e	$c(e)$
(D,E)	1
(D,L)	2
(F,J)	2
(G,J)	2
(C,D)	3
(E,F)	3
(H,I)	3
(A,B)	4
(B,C)	4
...	...

Exemplo



$$c(T) = 24$$

Componentes

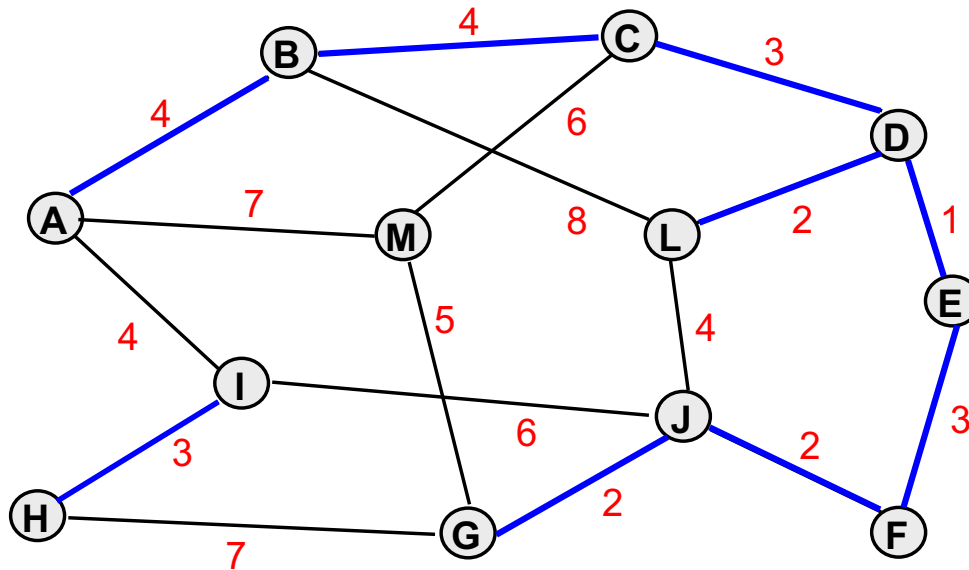
{ A, B, C, D, E, F, G, J, L }

{ H, I } { M }

**Lista
ordenada**

e	c(e)
...	...
(A,I)	4
(J,L)	4
(G,M)	5
(C,M)	6
(I,J)	6
(A,M)	7
(G,H)	7
(B,L)	8

Exemplo



$$c(T) = 24$$

Componentes

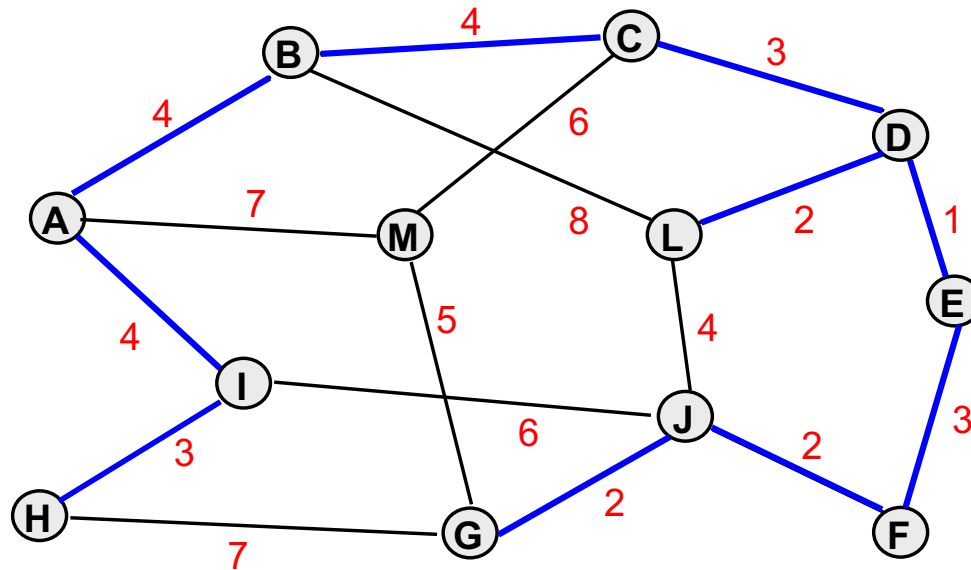
{ A, B, C, D, E, F, G, J, L }

{ H, I } { M }

**Lista
ordenada**

e	c(e)
...	...
(A,I)	4
(J,L)	4
(G,M)	5
(C,M)	6
(I,J)	6
(A,M)	7
(G,H)	7
(B,L)	8

Exemplo



$$c(T) = 24$$

Componentes

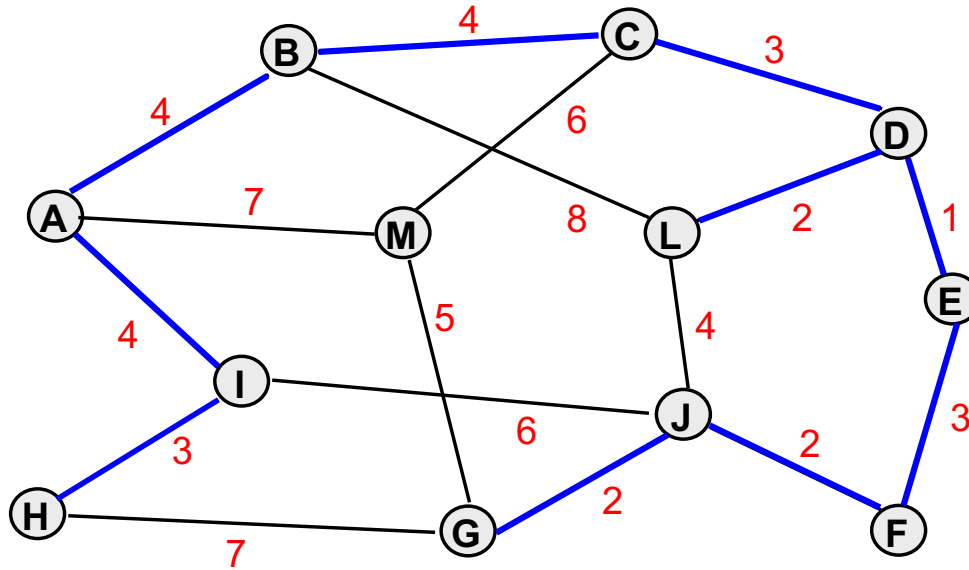
{ A, B, C, D, E, F, G, J, L }

{ H, I } { M }

**Lista
ordenada**

e	c(e)
...	...
(A,I)	4
(J,L)	4
(G,M)	5
(C,M)	6
(I,J)	6
(A,M)	7
(G,H)	7
(B,L)	8

Exemplo



$$c(T) = 24$$

Componentes

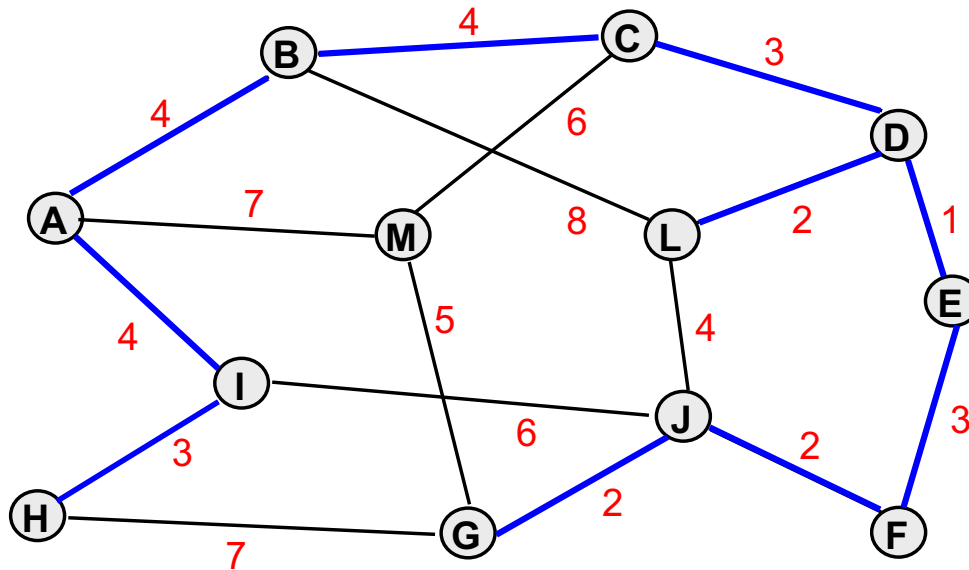
{ A, B, C, D, E, F, G, H, I, J, L }

{ M }

**Lista
ordenada**

e	c(e)
...	...
(A,I)	4
(J,L)	4
(G,M)	5
(C,M)	6
(I,J)	6
(A,M)	7
(G,H)	7
(B,L)	8

Exemplo



$$c(T) = 28$$

Componentes

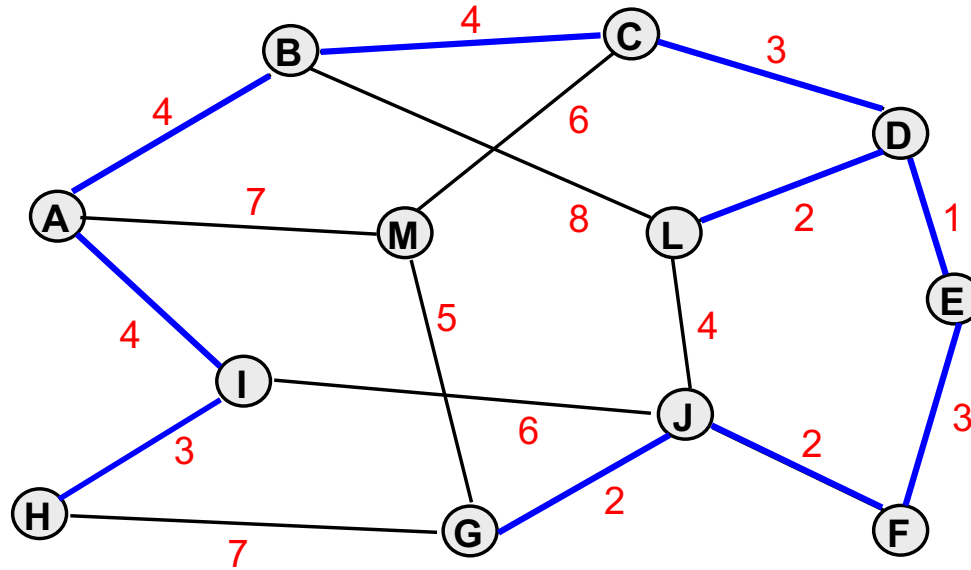
{ A, B, C, D, E, F, G, H, I, J, L }

{ M }

**Lista
ordenada**

e	c(e)
...	...
(A,I)	4
(J,L)	4
(G,M)	5
(C,M)	6
(I,J)	6
(A,M)	7
(G,H)	7
(B,L)	8

Exemplo



$$c(T) = 28$$

Componentes

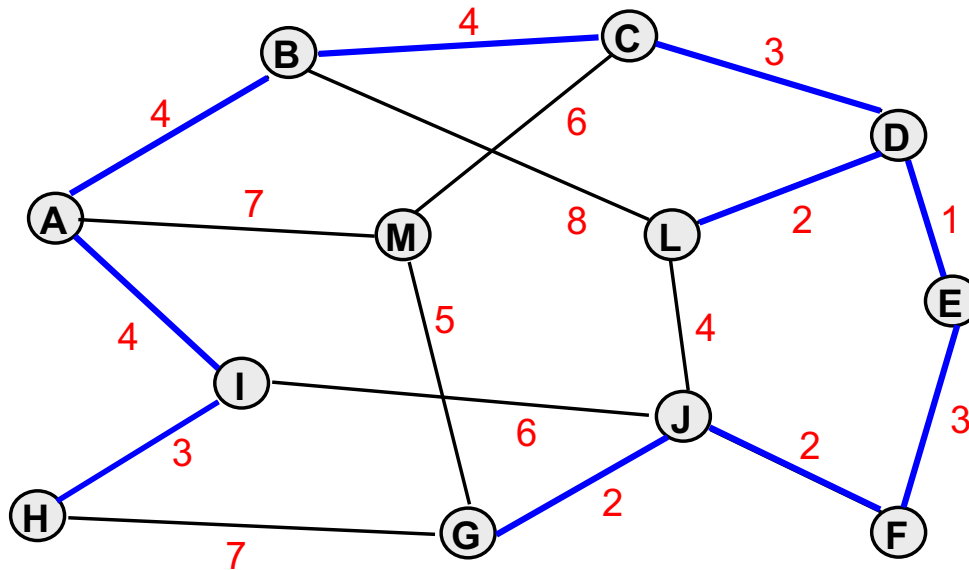
{ A, B, C, D, E, F, G, H, I, J, L }

{ M }

**Lista
ordenada**

e	c(e)
...	...
(A,I)	4
(J,L)	4
(G,M)	5
(C,M)	6
(I,J)	6
(A,M)	7
(G,H)	7
(B,L)	8

Exemplo



$$c(T) = 28$$

Componentes

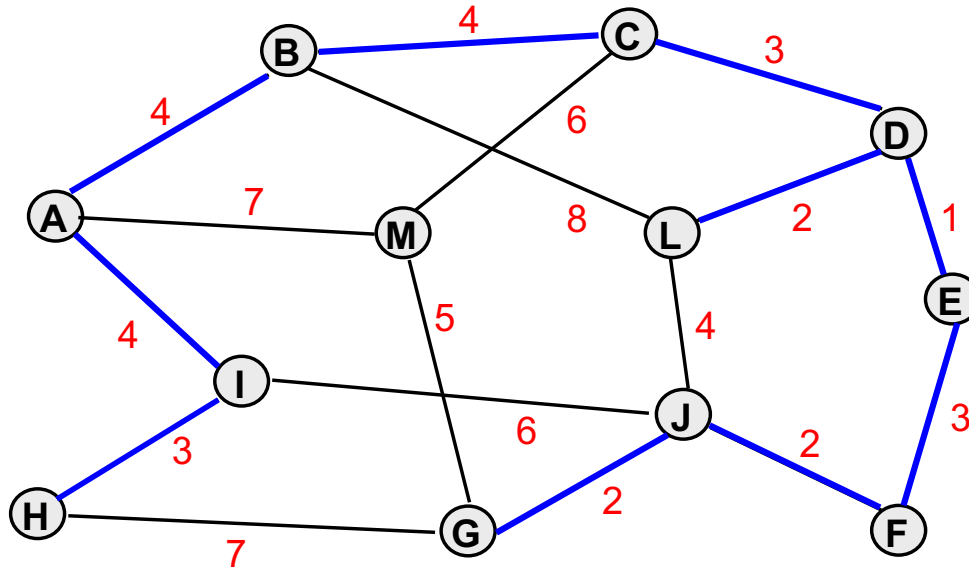
{ A, B, C, D, E, F, G, H, I, J, L }

{ M }

**Lista
ordenada**

e	c(e)
...	...
(A,I)	4
(J,L)	4
(G,M)	5
(C,M)	6
(I,J)	6
(A,M)	7
(G,H)	7
(B,L)	8

Exemplo



$c(T) = 28$

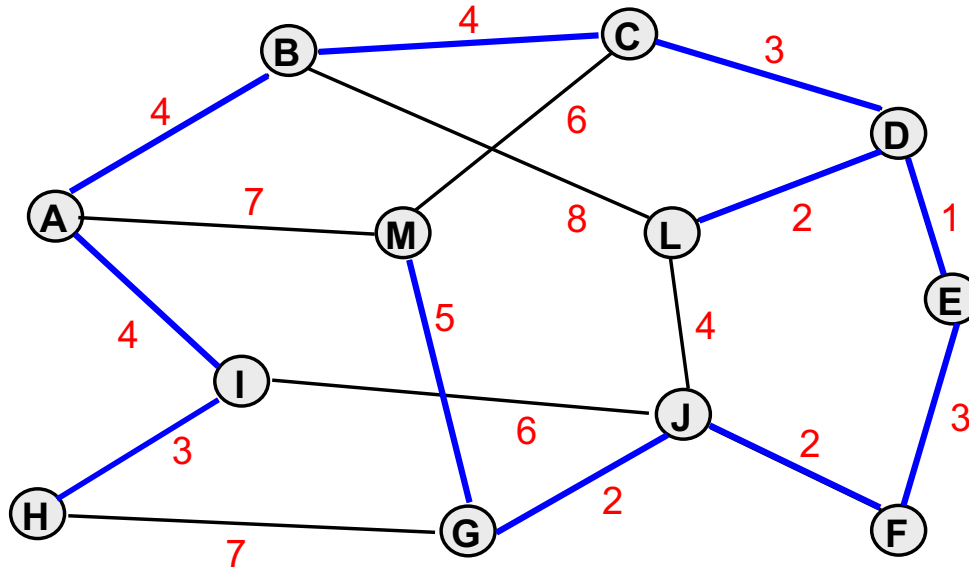
Componentes

{ A, B, C, D, E, F, G, H, I, J, L }
 { M }

Lista ordenada

e	$c(e)$
...	...
(A,I)	4
(J,L)	4
(G,M)	5
(C,M)	6
(I,J)	6
(A,M)	7
(G,H)	7
(B,L)	8

Exemplo



$$c(T) = 28$$

Componentes

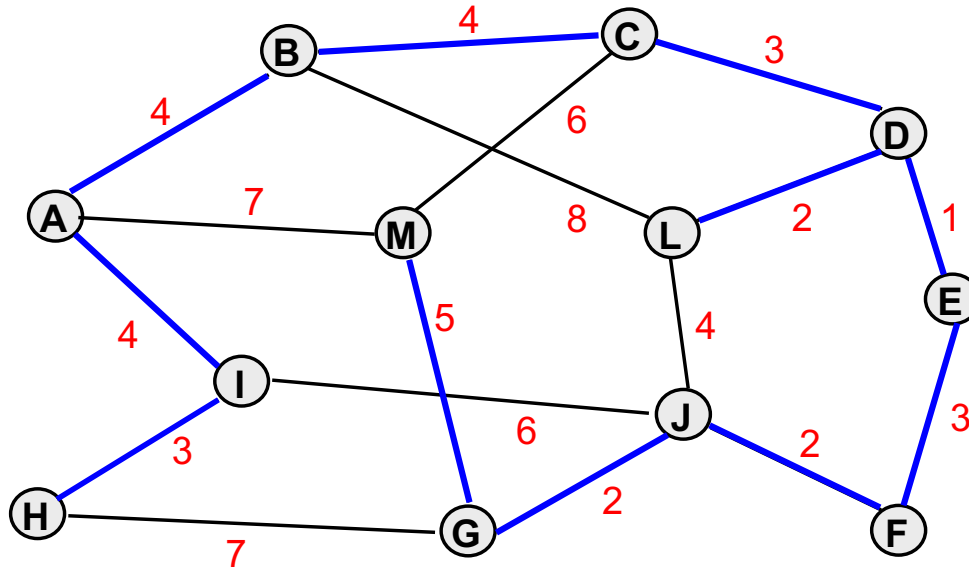
{ A, B, C, D, E, F, G, H, I, J, L }

{ M }

**Lista
ordenada**

e	c(e)
...	...
(A,I)	4
(J,L)	4
(G,M)	5
(C,M)	6
(I,J)	6
(A,M)	7
(G,H)	7
(B,L)	8

Exemplo



$$c(T) = 28$$

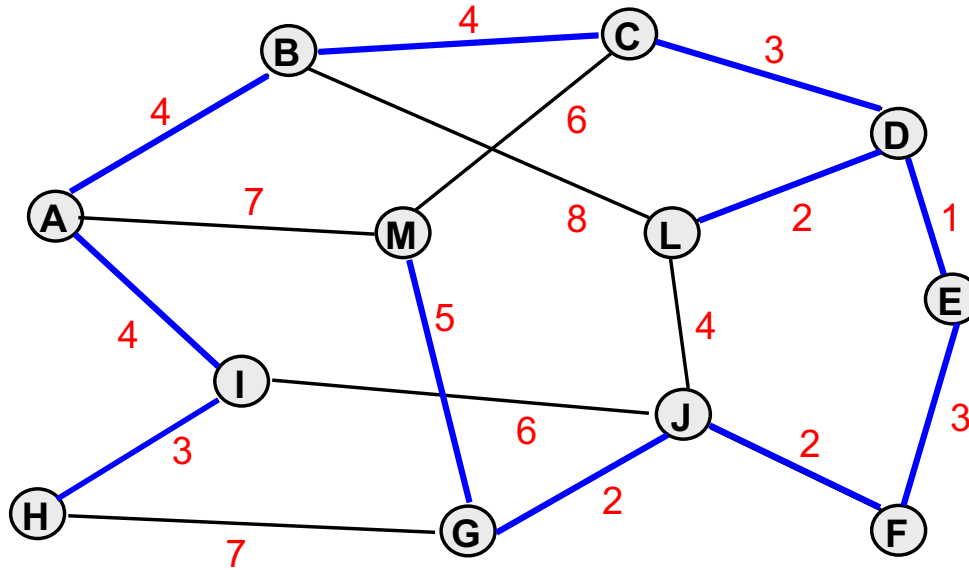
Componentes

{ A, B, C, D, E, F, G, H, I, J, L, M }

**Lista
ordenada**

e	c(e)
...	...
(A,I)	4
(J,L)	4
(G,M)	5
(C,M)	6
(I,J)	6
(A,M)	7
(G,H)	7
(B,L)	8

Exemplo



$$c(T) = 33$$

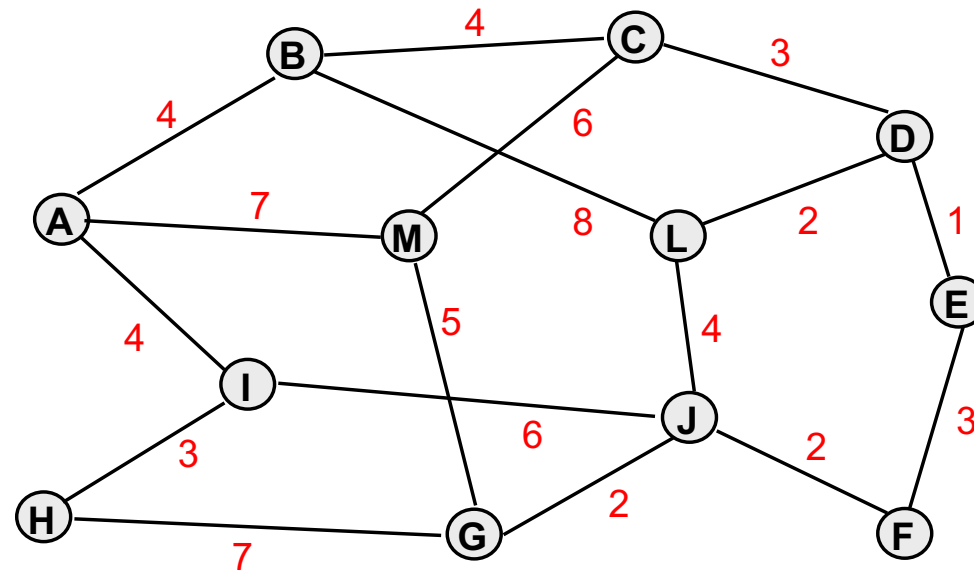
Componentes

{ A, B, C, D, E, F, G, H, I, J, L, M }

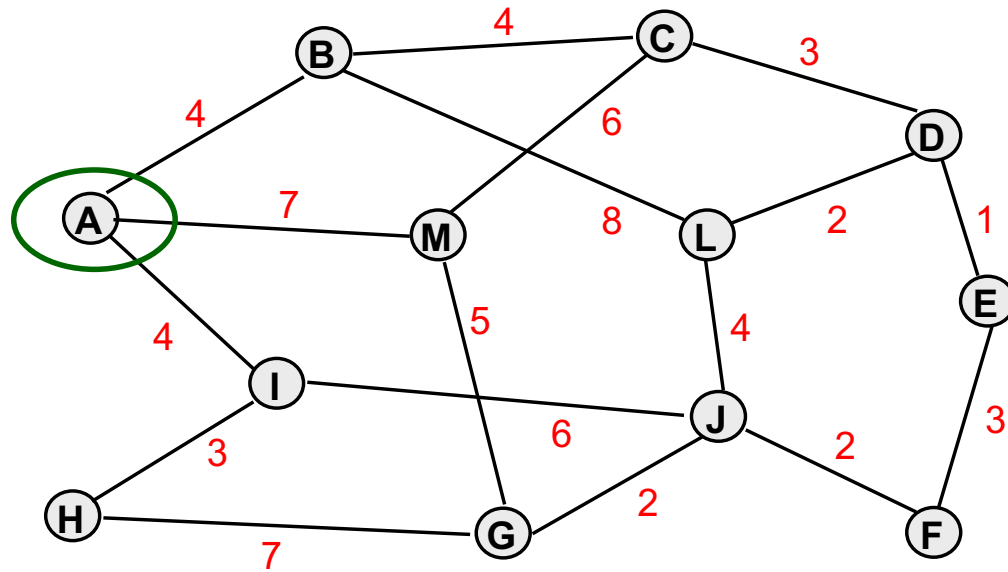
**Lista
ordenada**

e	c(e)
...	...
(A,I)	4
(J,L)	4
(G,M)	5
(C,M)	6
(I,J)	6
(A,M)	7
(G,H)	7
(B,L)	8

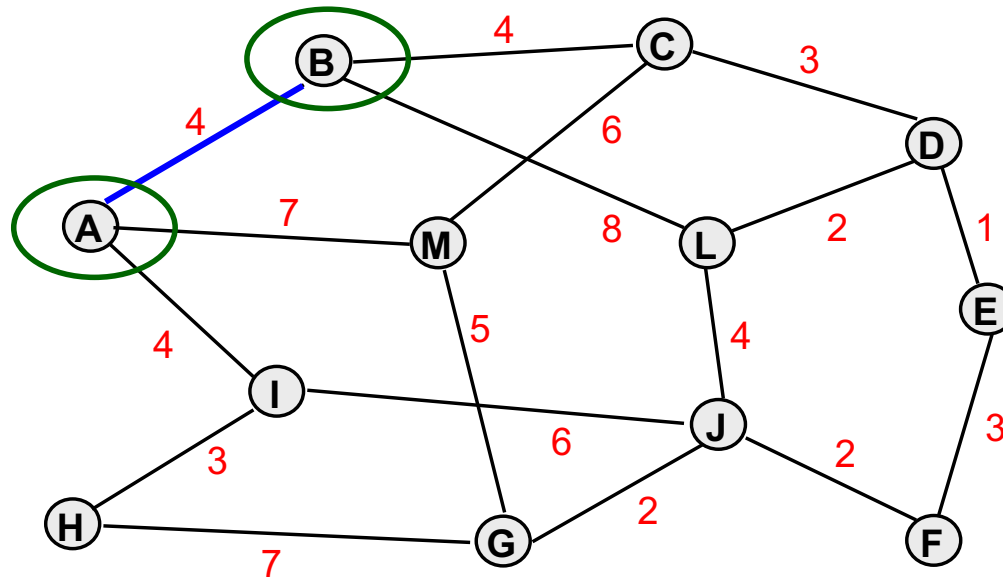
Exemplo



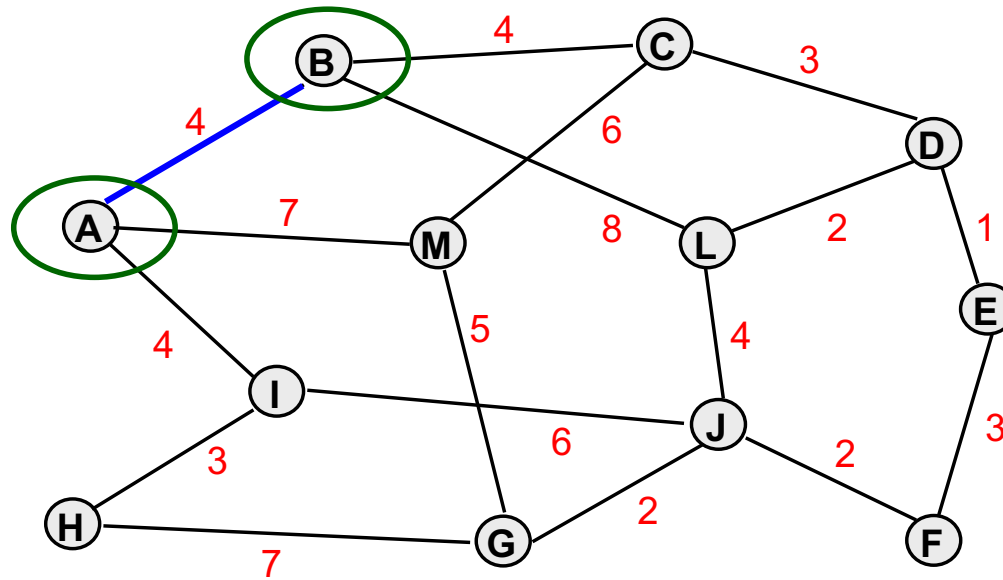
Exemplo



Exemplo

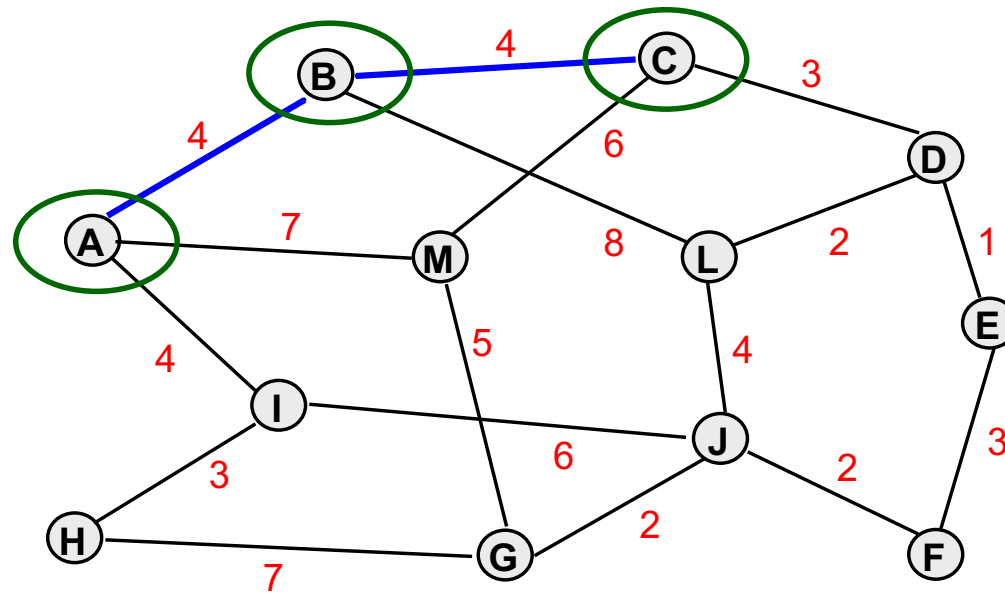


Exemplo



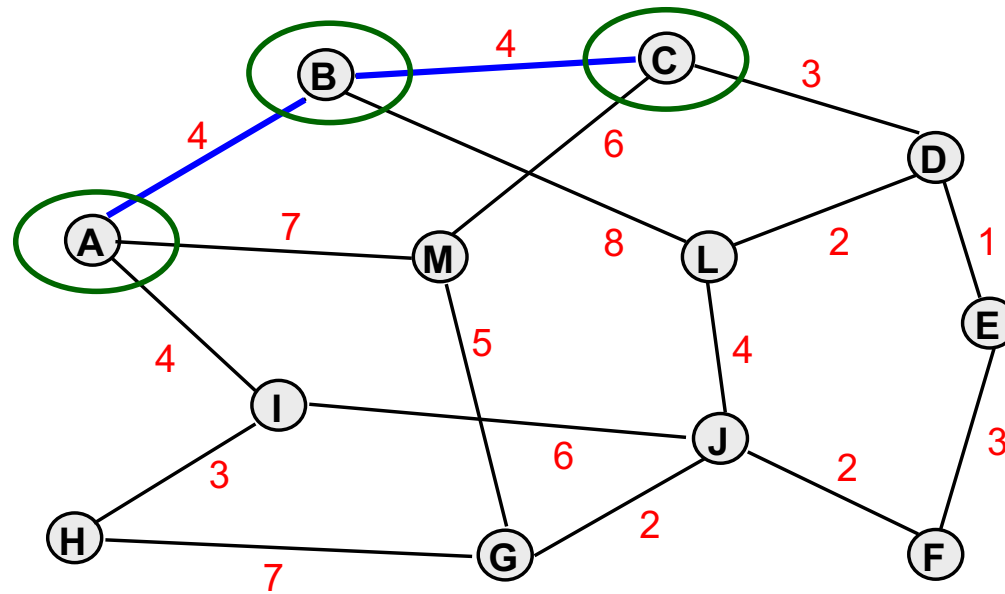
$$c(T) = 4$$

Exemplo



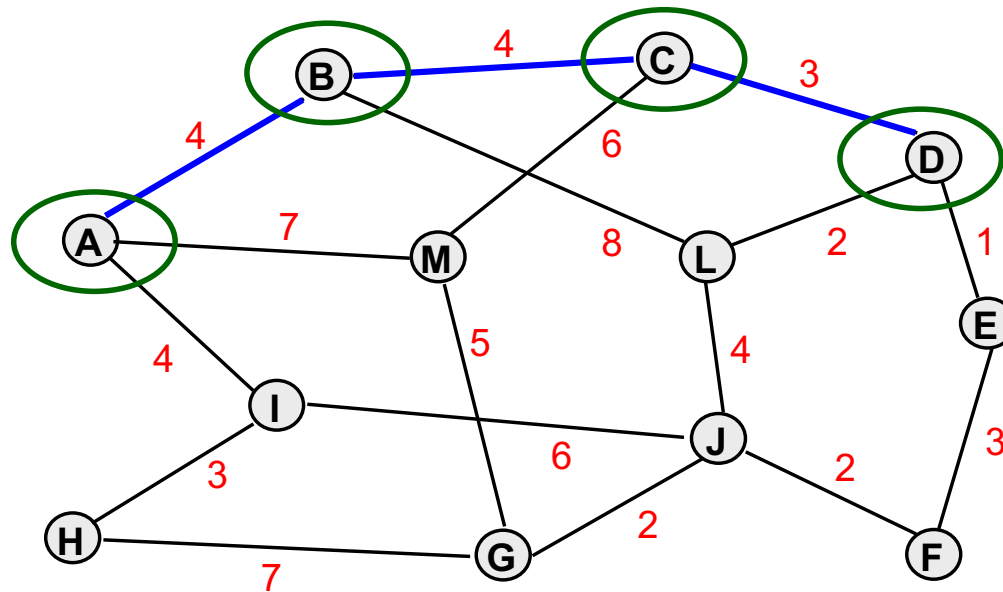
$$c(T) = 4$$

Exemplo



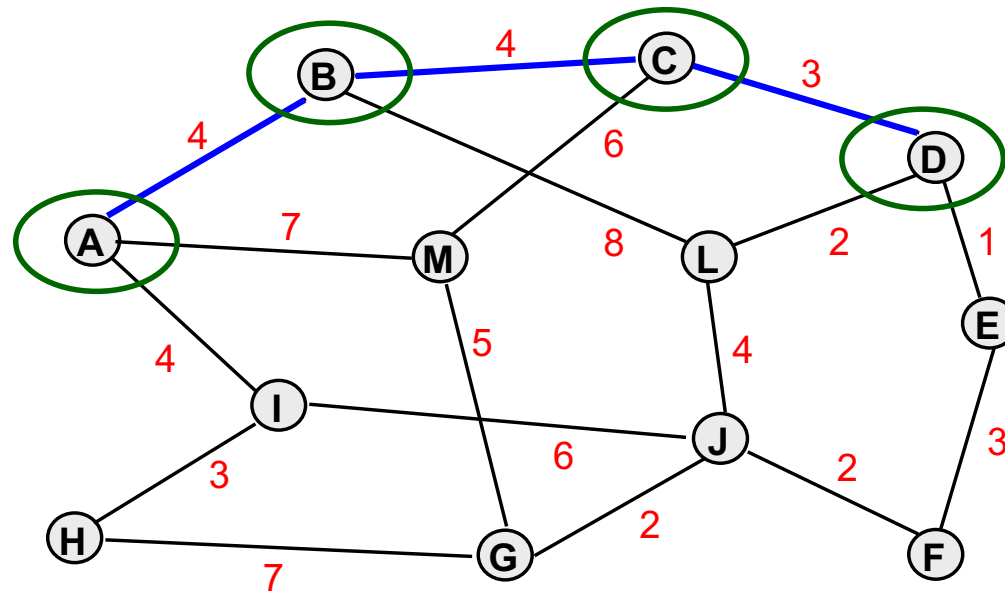
$$c(T) = 8$$

Exemplo



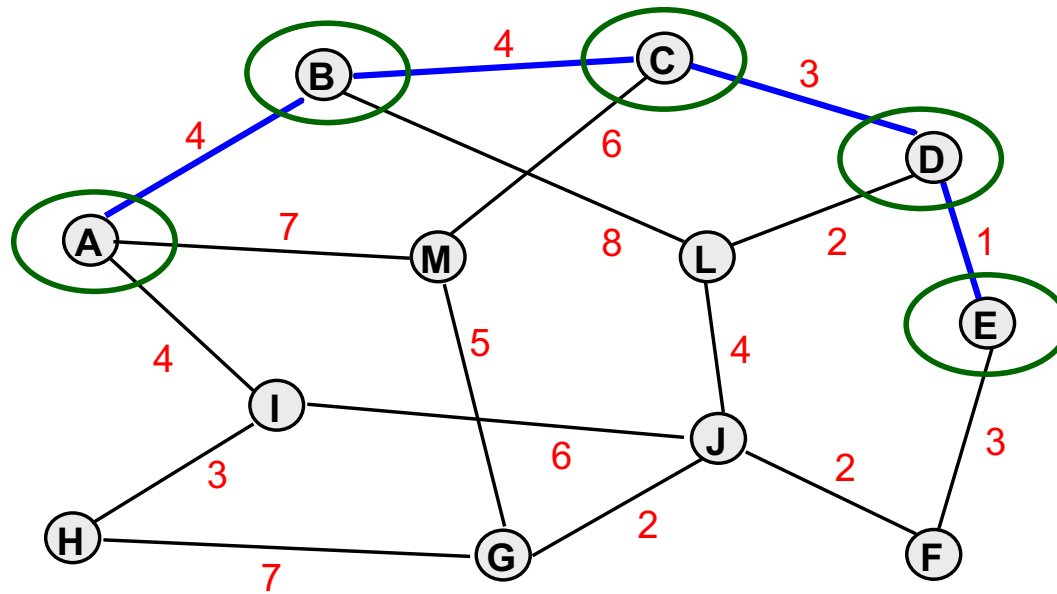
$$c(T) = 8$$

Exemplo



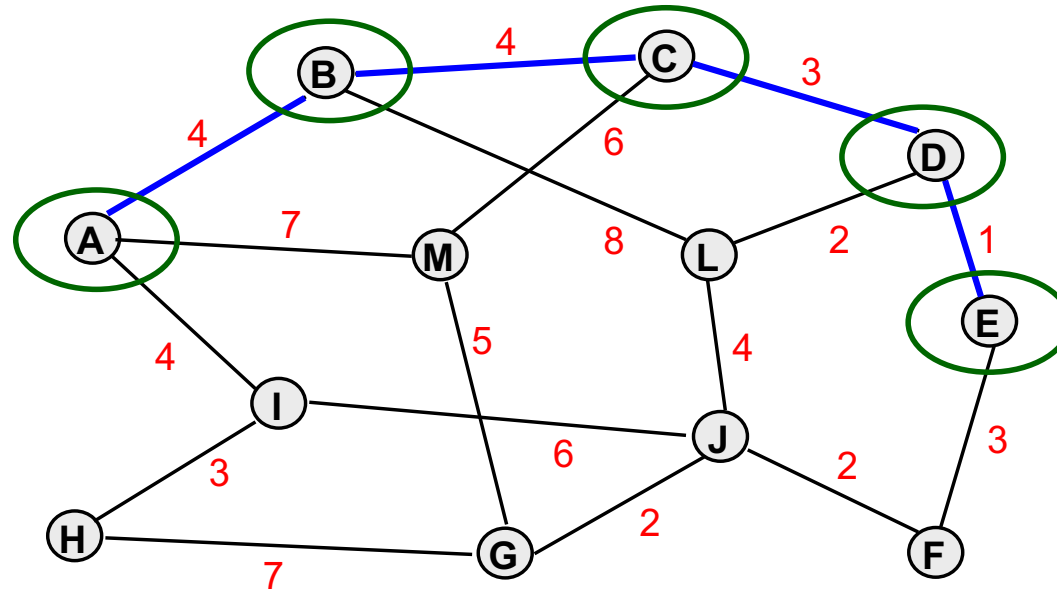
$$c(T) = 11$$

Exemplo



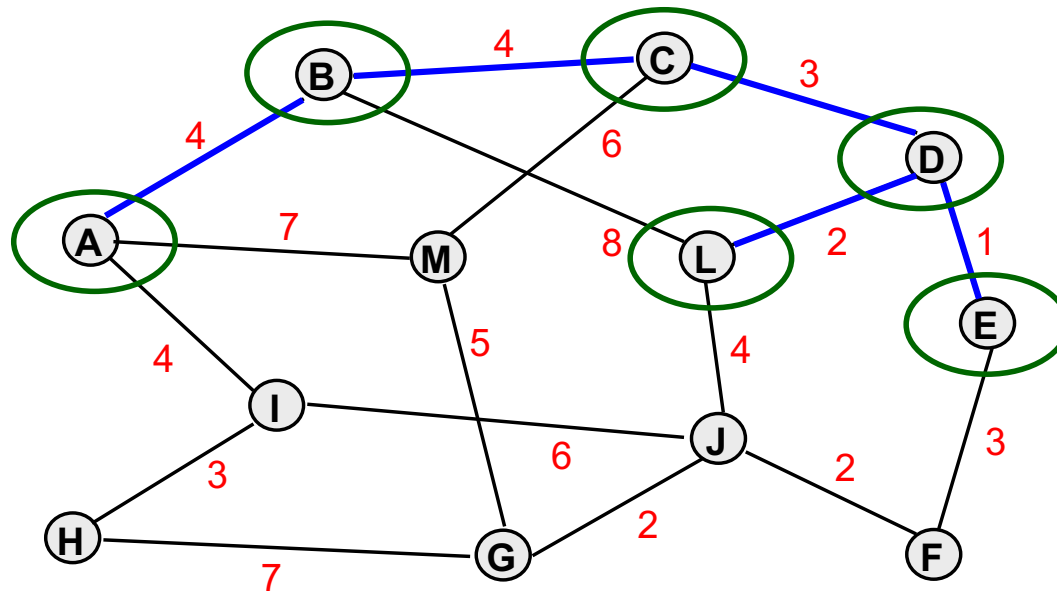
$$c(T) = 11$$

Exemplo



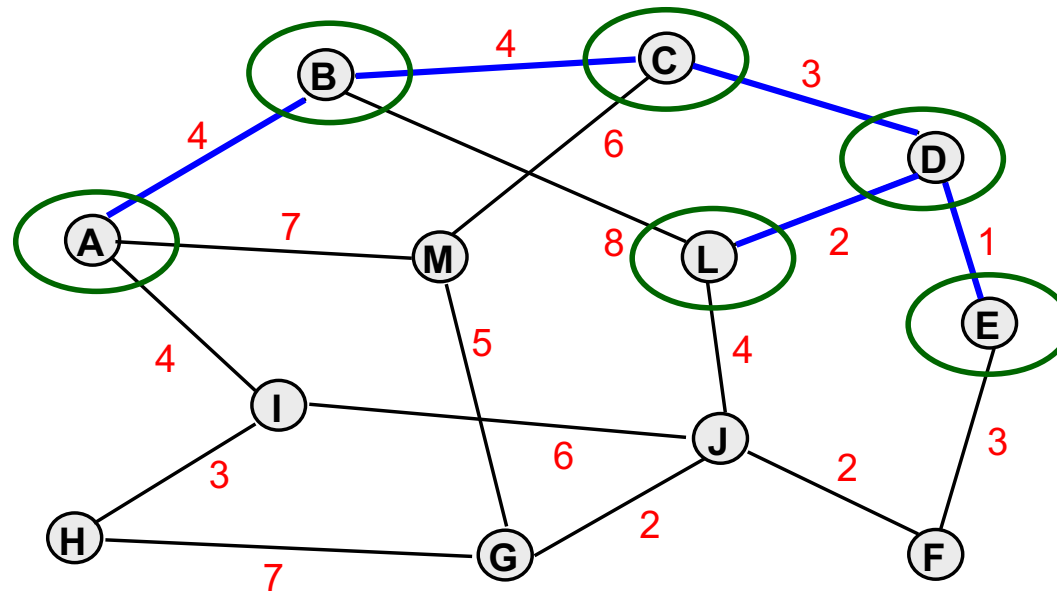
$$c(T) = 12$$

Exemplo



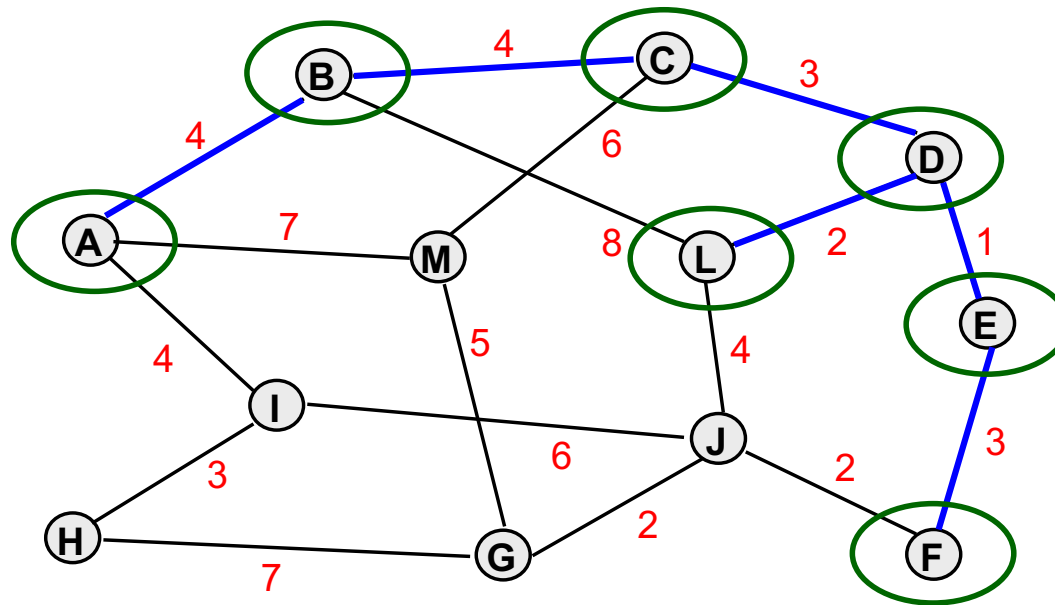
$$c(T) = 12$$

Exemplo



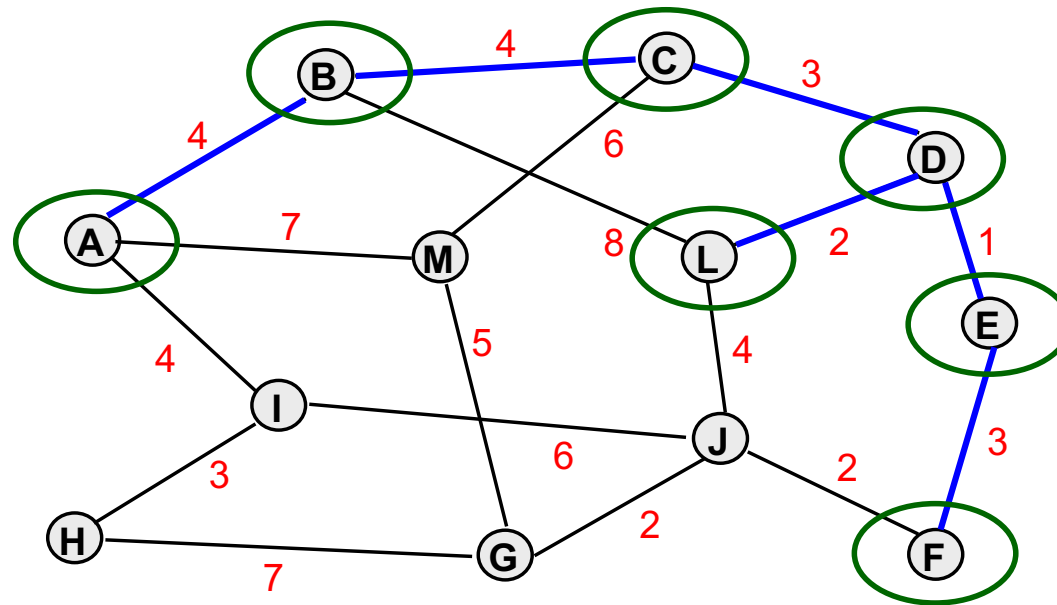
$$c(T) = 14$$

Exemplo



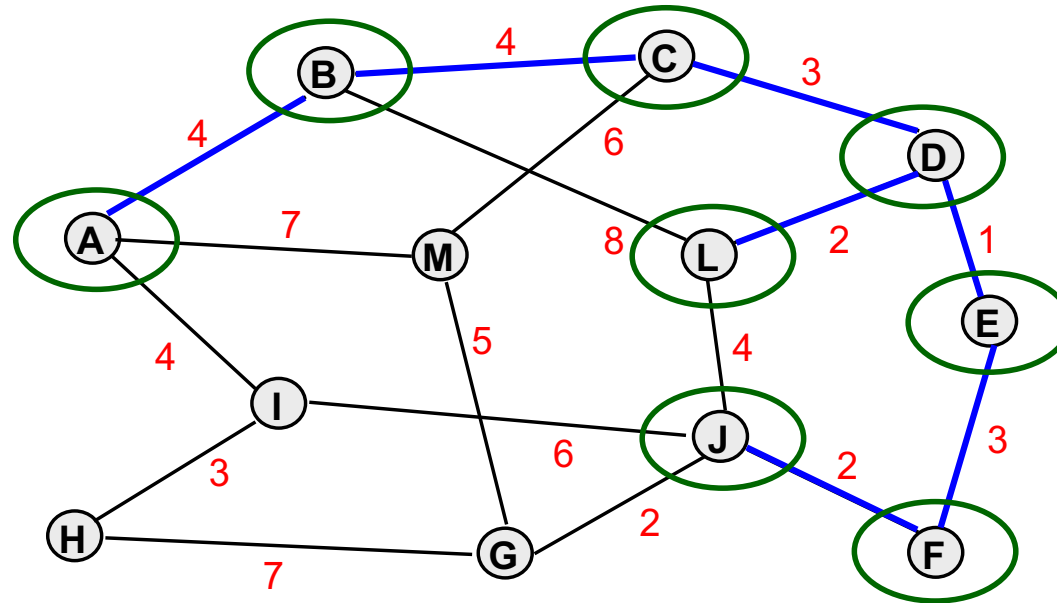
$$c(T) = 14$$

Exemplo



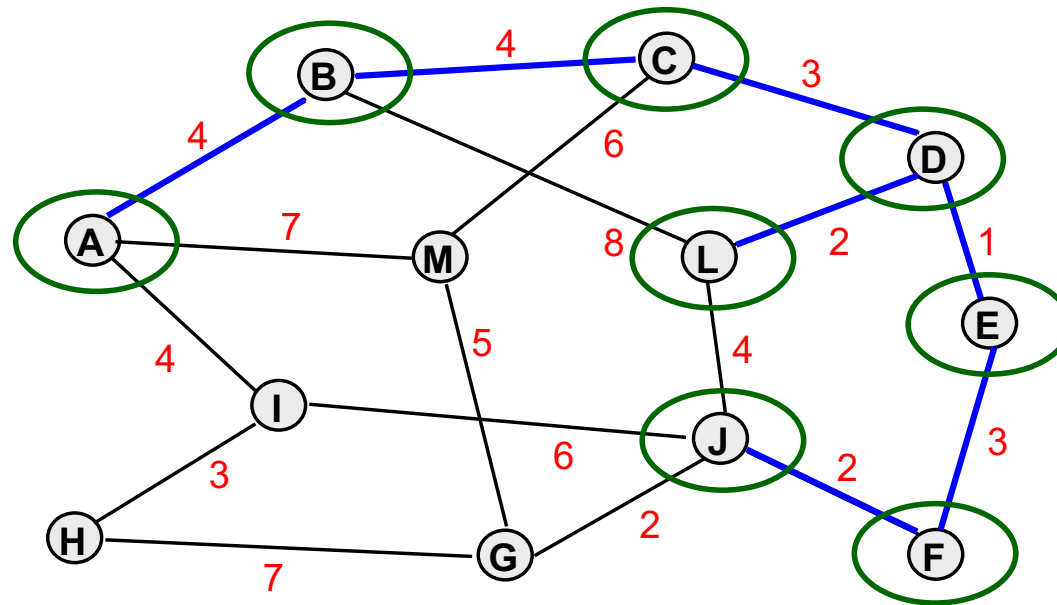
$$c(T) = 17$$

Exemplo



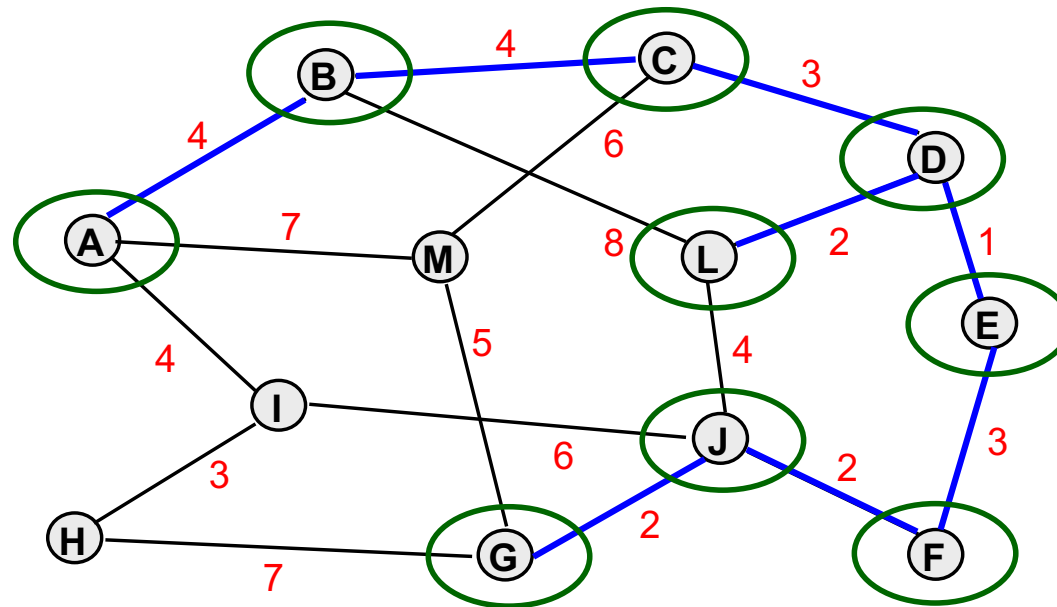
$$c(T) = 17$$

Exemplo



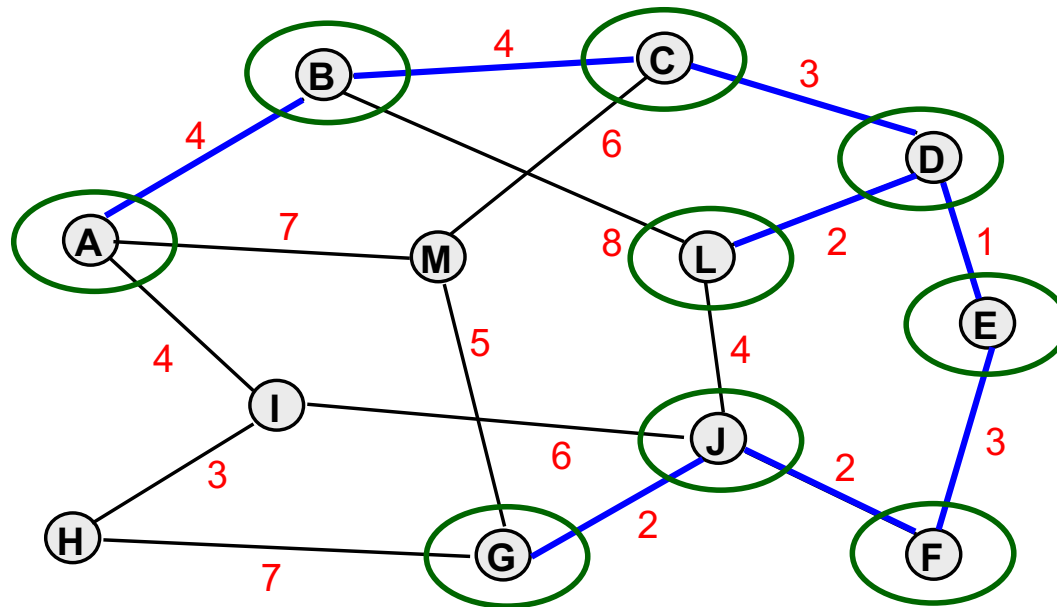
$$c(T) = 19$$

Exemplo



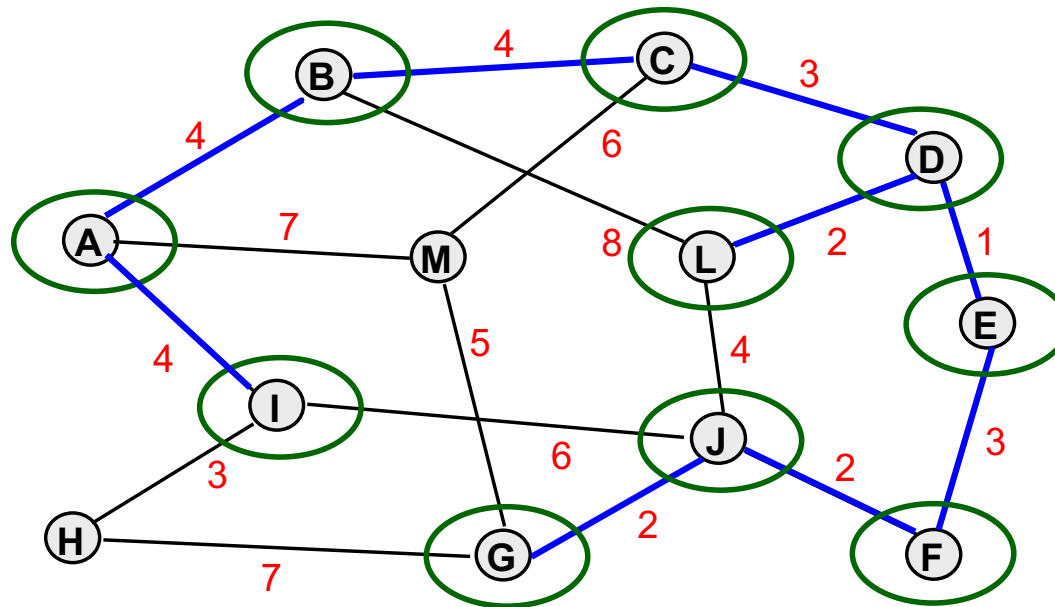
$$c(T) = 19$$

Exemplo



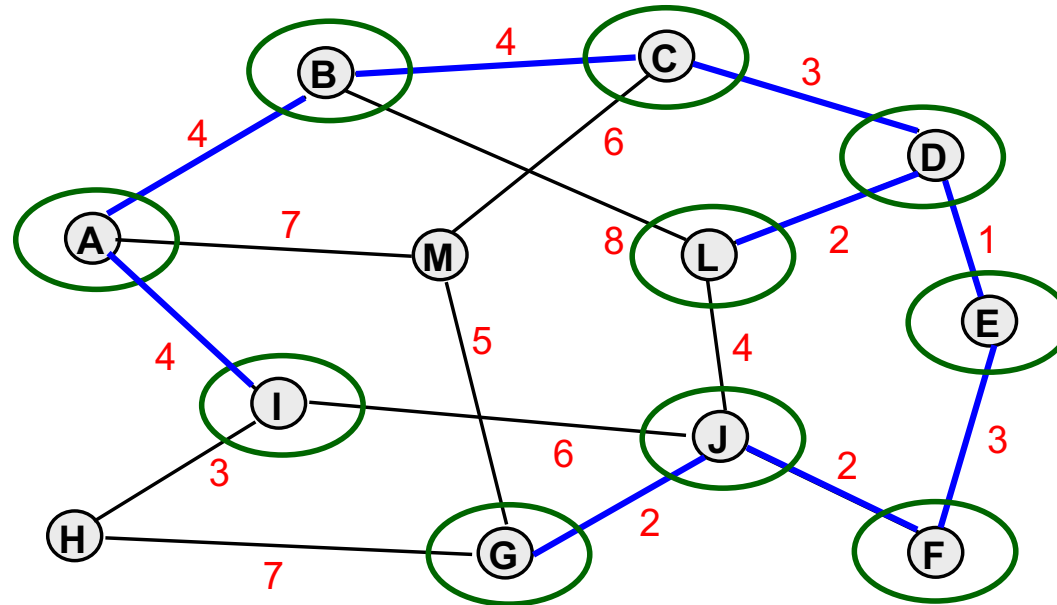
$$c(T) = 21$$

Exemplo



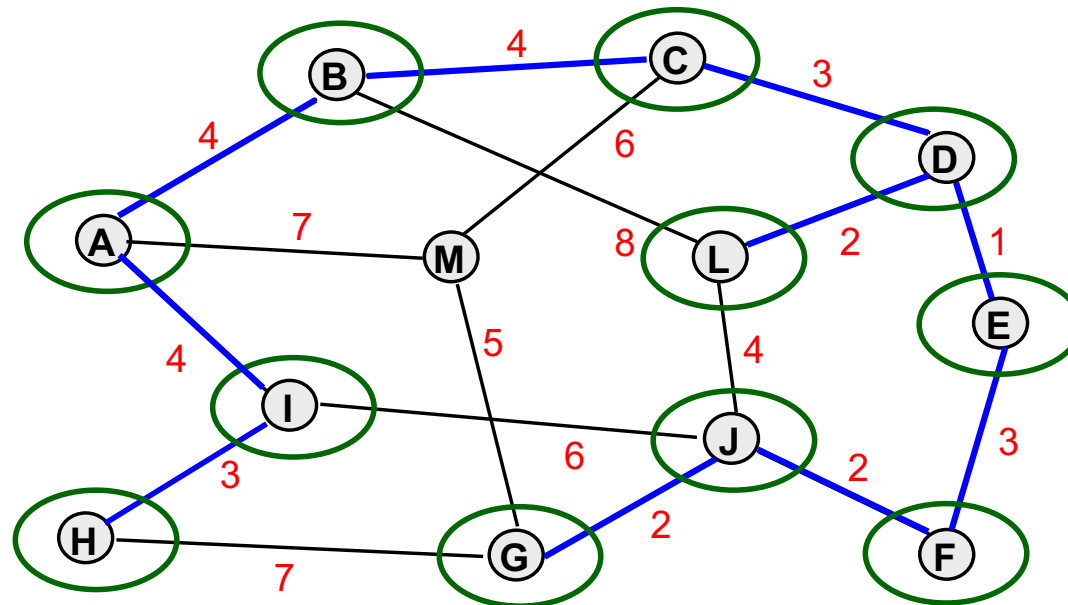
$$c(T) = 21$$

Exemplo



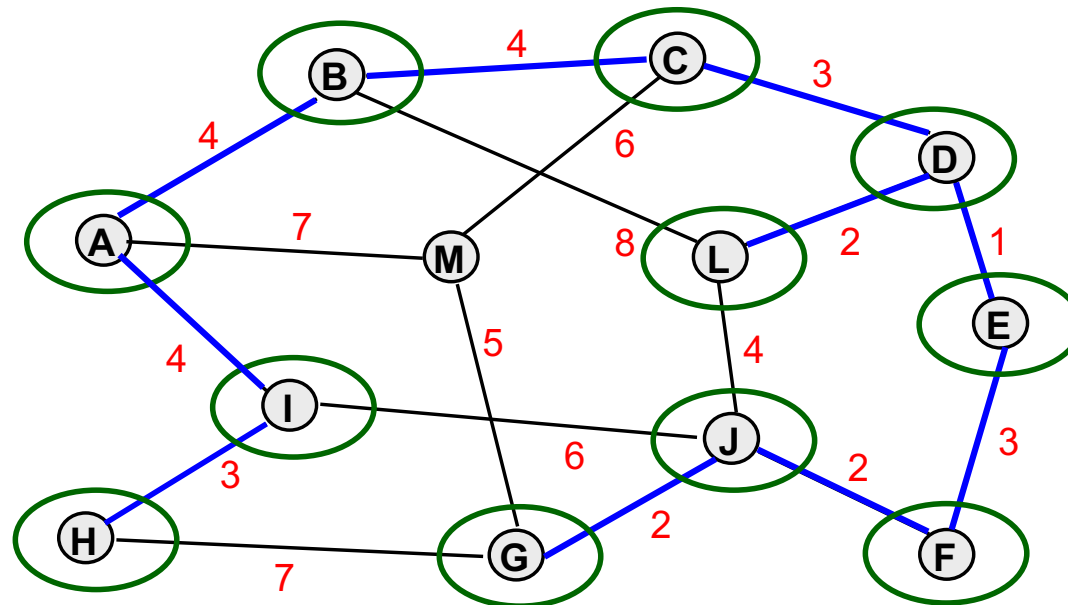
$$c(T) = 25$$

Exemplo



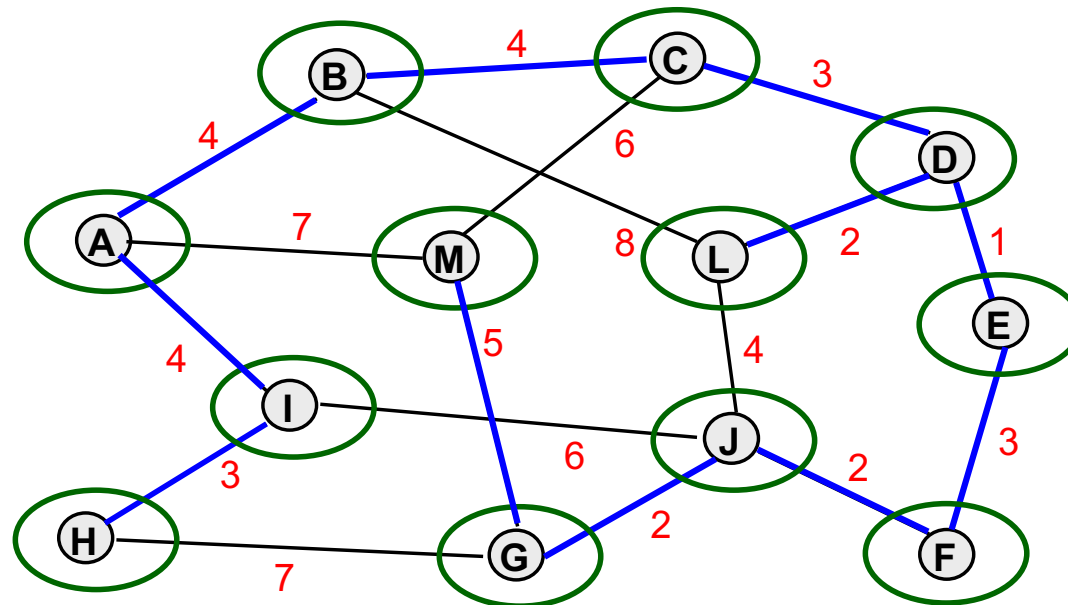
$$c(T) = 25$$

Exemplo



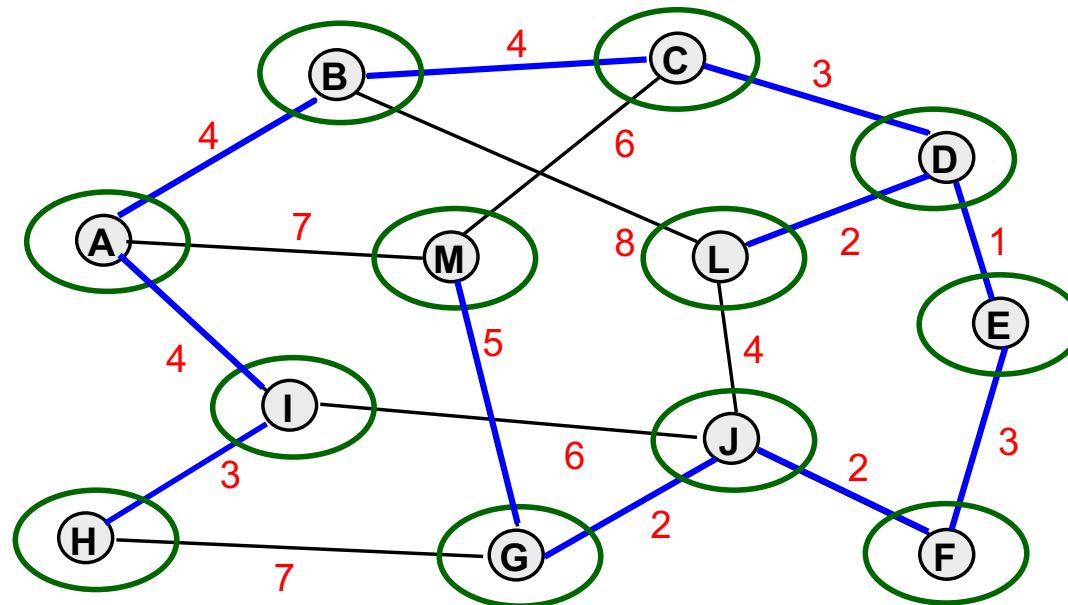
$$c(T) = 28$$

Exemplo



$$c(T) = 28$$

Exemplo



$$c(T) = 33$$

Moedas de troco



Moedas de troco



- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.

Moedas de troco



- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.

Moedas de troco



- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	8							
ultima	0	1	1	1	1	5	1	1	1							

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	4							
ultima	0	1	1	1	1	5	1	1	1							

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	4	9						
ultima	0	1	1	1	1	5	1	1	1	1						

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	4	5						
ultima	0	1	1	1	1	5	1	1	1	1						

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	4	5	10					
ultima	0	1	1	1	1	5	1	1	1	1	1					

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	4	5	6					
ultima	0	1	1	1	1	5	1	1	1	1	1					

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	4	5	2					
ultima	0	1	1	1	1	5	1	1	1	1	5					

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	4	5	1					
ultima	0	1	1	1	1	5	1	1	1	1	10					

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	4	5	1	11				
ultima	0	1	1	1	1	5	1	1	1	1	10	1				

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	4	5	1	2				
ultima	0	1	1	1	1	5	1	1	1	1	10	1				

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	4	5	1	2	12			
ultima	0	1	1	1	1	5	1	1	1	1	10	1	1			

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	4	5	1	2	3			
ultima	0	1	1	1	1	5	1	1	1	1	10	1	1			

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	4	5	1	2	3	13		
ultima	0	1	1	1	1	5	1	1	1	1	10	1	1	1		

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	4	5	1	2	3	4		
ultima	0	1	1	1	1	5	1	1	1	1	10	1	1	1		

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	4	5	1	2	3	4	14	
ultima	0	1	1	1	1	5	1	1	1	1	10	1	1	1	1	

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	4	5	1	2	3	4	5	
ultima	0	1	1	1	1	5	1	1	1	1	10	1	1	1	1	

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	4	5	1	2	3	4	5	15
ultima	0	1	1	1	1	5	1	1	1	1	10	1	1	1	1	1

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	4	5	1	2	3	4	5	6
ultima	0	1	1	1	1	5	1	1	1	1	10	1	1	1	1	1

Moedas de troco

- Quando ocorre sobreposição de subproblemas, é mais eficiente resolver todos os subproblemas em ordem incremental de tamanho, armazenando suas subsoluções em um vetor para eventuais consultas posteriores.
- Consideremos um exemplo: troco ótimo de 15 centavos utilizando moedas de 1, 5 e 10 centavos.
- Na simulação abaixo, supomos que o vetor de moedas esteja em ordem crescente, embora isso não seja necessário (nesse caso, seria encontrada outra solução também ótima).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
quant	0	1	2	3	4	1	2	3	4	5	1	2	3	4	5	2
ultima	0	1	1	1	1	5	1	1	1	1	10	1	1	1	1	5

Exemplo



Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0					
1					
2					
3					
4					

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0					
1					
2					
3					
4					

T	0	1	2	3	4
0					
1					
2					
3					
4					

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0				
1		0			
2			0		
3				0	
4					0

T	0	1	2	3	4
0					
1					
2					
3					
4					

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20			
1		0			
2			0		
3				0	
4					0

T	0	1	2	3	4
0					
1					
2					
3					
4					

$$\text{mín} \{ 0 + 0 + 5.4.1 \}$$

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20			
1		0			
2			0		
3				0	
4					0

T	0	1	2	3	4
0		0			
1					
2					
3					
4					

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20			
1		0	12		
2			0		
3				0	
4					0

T	0	1	2	3	4
0		0			
1					
2					
3					
4					

$$\text{mín} \{ 0 + 0 + 4.1.3 \}$$

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20			
1		0	12		
2			0		
3				0	
4					0

T	0	1	2	3	4
0		0			
1			1		
2					
3					
4					

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20			
1		0	12		
2			0	21	
3				0	
4					0

T	0	1	2	3	4
0		0			
1			1		
2					
3					
4					

$$\text{mín} \{ 0 + 0 + 1.3.7 \}$$

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20			
1		0	12		
2			0	21	
3				0	
4					0

T	0	1	2	3	4
0		0			
1			1		
2				2	
3					
4					

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20			
1		0	12		
2			0	21	
3				0	42
4					0

T	0	1	2	3	4
0		0			
1			1		
2				2	
3					
4					

$$\text{mín} \{ 0 + 0 + 3.7.2 \}$$

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20			
1		0	12		
2			0	21	
3				0	42
4					0

T	0	1	2	3	4
0		0			
1			1		
2				2	
3					3
4					

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20	35		
1		0	12		
2			0	21	
3				0	42
4					0

T	0	1	2	3	4
0		0			
1			1		
2				2	
3					3
4					

$$\text{mín} \{ 0 + 12 + 5.4.3, 20 + 0 + 5.1.3 \}$$

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20	35		
1		0	12		
2			0	21	
3				0	42
4					0

T	0	1	2	3	4
0		0	1		
1			1		
2				2	
3					3
4					

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20	35		
1		0	12	49	
2			0	21	
3				0	42
4					0

T	0	1	2	3	4
0		0	1		
1			1		
2				2	
3					3
4					

$$\text{mín} \{ 0 + 21 + 4.1.7, 12 + 0 + 4.3.7 \}$$

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20	35		
1		0	12	49	
2			0	21	
3				0	42
4					0

T	0	1	2	3	4
0		0	1		
1			1	1	
2				2	
3					3
4					

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20	35		
1		0	12	49	
2			0	21	35
3				0	42
4					0

T	0	1	2	3	4
0		0	1		
1			1	1	
2				2	
3					3
4					

$$\text{mín} \{ 0 + 42 + 1.3.2, 21 + 0 + 1.7.2 \}$$

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20	35		
1		0	12	49	
2			0	21	35
3				0	42
4					0

T	0	1	2	3	4
0		0	1		
1			1	1	
2				2	3
3					3
4					

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20	35	76	
1		0	12	49	
2			0	21	35
3				0	42
4					0

T	0	1	2	3	4
0		0	1		
1			1	1	
2				2	3
3					3
4					

$$\text{mín} \{ 0 + 49 + 5.4.7, 20 + 21 + 5.1.7, 35 + 0 + 5.3.7 \}$$

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20	35	76	
1		0	12	49	
2			0	21	35
3				0	42
4					0

T	0	1	2	3	4
0		0	1	1	
1			1	1	
2				2	3
3					3
4					

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20	35	76	
1		0	12	49	43
2			0	21	35
3				0	42
4					0

T	0	1	2	3	4
0		0	1	1	
1			1	1	
2				2	3
3					3
4					

$$\text{mín} \{ 0 + 35 + 4.1.2, 12 + 42 + 4.3.2, 49 + 0 + 4.7.2 \}$$

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20	35	76	
1		0	12	49	43
2			0	21	35
3				0	42
4					0

T	0	1	2	3	4
0		0	1	1	
1			1	1	1
2				2	3
3					3
4					

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20	35	76	65
1		0	12	49	43
2			0	21	35
3				0	42
4					0

T	0	1	2	3	4
0		0	1	1	
1			1	1	1
2				2	3
3					3
4					

$$\text{mín} \{ 0 + 43 + 5.4.2, 20 + 35 + 5.1.2, 35 + 42 + 5.3.2, 76 + 0 + 5.7.2 \}$$

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20	35	76	65
1		0	12	49	43
2			0	21	35
3				0	42
4					0

T	0	1	2	3	4
0		0	1	1	1
1			1	1	1
2				2	3
3					3
4					

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20	35	76	65
1		0	12	49	43
2			0	21	35
3				0	42
4					0

T	0	1	2	3	4
0		0	1	1	1
1			1	1	1
2				2	3
3					3
4					

Sequência ótima: $A_0 A_1 A_2 A_3 A_4$

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20	35	76	65
1		0	12	49	43
2			0	21	35
3				0	42
4					0

T	0	1	2	3	4
0		0	1	1	1
1			1	1	1
2				2	3
3					3
4					

Sequência ótima: $(A_0 A_1 . A_2 A_3 A_4)$

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20	35	76	65
1		0	12	49	43
2			0	21	35
3				0	42
4					0

T	0	1	2	3	4
0		0	1	1	1
1			1	1	1
2				2	3
3					3
4					

Sequência ótima: $((A_0.A_1).A_2.A_3.A_4)$

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20	35	76	65
1		0	12	49	43
2			0	21	35
3				0	42
4					0

T	0	1	2	3	4
0		0	1	1	1
1			1	1	1
2				2	3
3					3
4					

Sequência ótima: $((A_0.A_1).(A_2.A_3.A_4))$

Exemplo

- $A = A_0.A_1.A_2.A_3.A_4$, onde as dimensões são, respectivamente: 5×4 , 4×1 , 1×3 , 3×7 e 7×2 .
- Portanto: $d_0=5$, $d_1=4$, $d_2=1$, $d_3=3$, $d_4=7$ e $d_5=2$.

N	0	1	2	3	4
0	0	20	35	76	65
1		0	12	49	43
2			0	21	35
3				0	42
4					0

T	0	1	2	3	4
0		0	1	1	1
1			1	1	1
2				2	3
3					3
4					

Sequência ótima: $((A_0 . A_1) . ((A_2 . A_3) . A_4))$

Exemplo



$X = \langle GGATCGA \rangle$ e $Y = \langle GAATTCAGTTA \rangle$

Exemplo

$X = \langle GGATCGA \rangle$ e $Y = \langle GAATTCAGTTA \rangle$

	G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1	1
G	0	1	1	1	1	1	1	2	2	2	2
A	0	1	2	2	2	2	2	2	2	2	3
T	0	1	2	2	3	3	3	3	3	3	3
C	0	1	2	2	3	3	4	4	4	4	4
G	0	1	2	2	3	3	4	4	5	5	5
A	0	1	2	3	3	3	4	5	5	5	6

Exemplo

$X = \langle GGATCGA \rangle$ e $Y = \langle GAATTCAGTTA \rangle$

	G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1	1
G	0	1	1	1	1	1	1	2	2	2	2
A	0	1	2	2	2	2	2	2	2	2	3
T	0	1	2	2	3	3	3	3	3	3	3
C	0	1	2	2	3	3	4	4	4	4	4
G	0	1	2	2	3	3	4	4	5	5	5
A	0	1	2	3	3	3	4	5	5	5	6

$|LCS| = 6$

Exemplo

$X = \langle GGATCGA \rangle$ e $Y = \langle GAATTCAGTTA \rangle$

	G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1	1
G	0	1	1	1	1	1	1	2	2	2	2
A	0	1	2	2	2	2	2	2	2	2	3
T	0	1	2	2	3	3	3	3	3	3	3
C	0	1	2	2	3	3	4	4	4	4	4
G	0	1	2	2	3	3	4	4	5	5	5
A	0	1	2	3	3	3	4	5	5	5	6

$|LCS| = 6$

Exemplo

$X = \langle GGATCGA \rangle$ e $Y = \langle GAATTCAGTTA \rangle$

	G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1	1
G	0	1	1	1	1	1	1	2	2	2	2
A	0	1	2	2	2	2	2	2	2	2	3
T	0	1	2	2	3	3	3	3	3	3	3
C	0	1	2	2	3	3	4	4	4	4	4
G	0	1	2	2	3	3	4	4	5	5	5
A	0	1	2	3	3	3	4	5	5	5	6

$|LCS| = 6$

Exemplo

$X = \langle GGATCGA \rangle$ e $Y = \langle GAATTCAGTTA \rangle$

	G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1	1
G	0	1	1	1	1	1	1	2	2	2	2
A	0	1	2	2	2	2	2	2	2	2	3
T	0	1	2	2	3	3	3	3	3	3	3
C	0	1	2	2	3	3	4	4	4	4	4
G	0	1	2	2	3	3	4	4	5	→ 5	→ 5
A	0	1	2	3	3	3	4	5	5	5	↘ 6

$|LCS| = 6$

Exemplo

$X = \langle GGATCGA \rangle$ e $Y = \langle GAATTCAGTTA \rangle$

	G	A	A	T	T	C	A	G	T	T	A	
	0	0	0	0	0	0	0	0	0	0	0	
G	0	1	1	1	1	1	1	1	1	1	1	
G	0	1	1	1	1	1	1	2	2	2	2	
A	0	1	2	2	2	2	2	2	2	2	3	
T	0	1	2	2	3	3	3	3	3	3	3	
C	0	1	2	2	3	3	4	4	4	4	4	
G	0	1	2	2	3	3	4	4	5	5	5	5
A	0	1	2	3	3	3	4	5	5	5	5	6

$|LCS| = 6$

Exemplo

$X = \langle GGATCGA \rangle$ e $Y = \langle GAATTCAGTTA \rangle$

	G	A	A	T	T	C	A	G	T	T	A	
	0	0	0	0	0	0	0	0	0	0	0	
G	0	1	1	1	1	1	1	1	1	1	1	
G	0	1	1	1	1	1	1	2	2	2	2	
A	0	1	2	2	2	2	2	2	2	2	3	
T	0	1	2	2	3	3	3	3	3	3	3	
C	0	1	2	2	3	3	4	4 →	4	4	4	
G	0	1	2	2	3	3	4	4	5 →	5 →	5 →	5
A	0	1	2	3	3	3	4	5	5	5	5	6 ↘

$|LCS| = 6$

Exemplo

$X = \langle GGATCGA \rangle$ e $Y = \langle GAATTCAGTTA \rangle$

	G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1	1
G	0	1	1	1	1	1	1	2	2	2	2
A	0	1	2	2	2	2	2	2	2	2	3
T	0	1	2	2	3	3	3	3	3	3	3
C	0	1	2	2	3	3	4	4	4	4	4
G	0	1	2	2	3	3	4	4	5	5	5
A	0	1	2	3	3	3	4	5	5	5	6

$|LCS| = 6$

Exemplo

$X = \langle GGATCGA \rangle$ e $Y = \langle GAATTCAGTTA \rangle$

	G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1	1
G	0	1	1	1	1	1	1	2	2	2	2
A	0	1	2	2	2	2	2	2	2	2	3
T	0	1	2	2	3	3	3	3	3	3	3
C	0	1	2	2	3	3	4	4	4	4	4
G	0	1	2	2	3	3	4	4	5	5	5
A	0	1	2	3	3	3	4	5	5	5	6

$|LCS| = 6$

Exemplo

$X = \langle GGATCGA \rangle$ e $Y = \langle GAATTCAGTTA \rangle$

	G	A	A	T	T	C	A	G	T	T	A		
	0	0	0	0	0	0	0	0	0	0	0		
G	0	1	1	1	1	1	1	1	1	1	1		
G	0	1	1	1	1	1	1	2	2	2	2		
A	0	1	2	2	→	2	2	2	2	2	3		
T	0	1	2	2	3	↘	3	3	3	3	3		
C	0	1	2	2	3	3	↘	↘	4	4	4		
G	0	1	2	2	3	3	4	4	↘	↘	↘	5	
A	0	1	2	3	3	3	4	5	5	5	5	↘	6

$|LCS| = 6$

Exemplo

$X = \langle GGATCGA \rangle$ e $Y = \langle GAATTCAGTTA \rangle$

	G	A	A	T	T	C	A	G	T	T	A	
	0	0	0	0	0	0	0	0	0	0	0	
G	0	1	1	1	1	1	1	1	1	1	1	
G	0	1	1	1	1	1	1	2	2	2	2	
A	0	1	2	2 →	→ 2	2	2	2	2	2	3	
T	0	1	2	2	3	3	3	3	3	3	3	
C	0	1	2	2	3	3	4 →	→ 4	4	4	4	
G	0	1	2	2	3	3	4	4	5 →	→ 5	→ 5	5
A	0	1	2	3	3	3	4	5	5	5	5	6

$|LCS| = 6$

Exemplo

$X = \langle GGATCGA \rangle$ e $Y = \langle GAATTCAGTTA \rangle$

	G	A	A	T	T	C	A	G	T	T	A		
	0	0	0	0	0	0	0	0	0	0	0		
G	0	1	1	1	1	1	1	1	1	1	1		
G	0	1	↓	1	1	1	1	2	2	2	2		
A	0	1	2	↘	→	2	2	2	2	2	3		
T	0	1	2	2	3	↘	3	3	3	3	3		
C	0	1	2	2	3	3	↘	→	4	4	4		
G	0	1	2	2	3	3	4	4	↘	→	→	5	
A	0	1	2	3	3	3	4	5	5	5	5	↘	6

$|LCS| = 6$

Exemplo

$X = \langle GGATCGA \rangle$ e $Y = \langle GAATTCAGTTA \rangle$

		G	A	A	T	T	C	A	G	T	T	A	
	0	0	0	0	0	0	0	0	0	0	0	0	
G	0	0	1	1 →	1	1	1	1	1	1	1	1	
G	0	0	1	1 ↓	1	1	1	1	2	2	2	2	
A	0	0	1	2	2 ↘	2 →	2	2	2	2	2	3	
T	0	0	1	2	2	3	3 ↘	3	3	3	3	3	
C	0	0	1	2	2	3	3	4 ↘	4 →	4	4	4	
G	0	0	1	2	2	3	3	4	4	5 ↘	5 →	5 →	
A	0	0	1	2	3	3	3	4	5	5	5	5	6 ↘

$|LCS| = 6$

Exemplo

$X = \langle GGATCGA \rangle$ e $Y = \langle GAATTCAGTTA \rangle$

		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1	1	1
G	0	1	1	1	1	1	1	1	2	2	2	2
A	0	1	2	2	2	2	2	2	2	2	2	3
T	0	1	2	2	3	3	3	3	3	3	3	3
C	0	1	2	2	3	3	4	4	4	4	4	4
G	0	1	2	2	3	3	4	4	5	5	5	5
A	0	1	2	3	3	3	4	5	5	5	5	6

$|LCS| = 6$

Exemplo

$X = \langle GGATCGA \rangle$ e $Y = \langle GAATTCAGTTA \rangle$

		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1	1	1
G	0	1	1	1	1	1	1	1	2	2	2	2
A	0	1	2	2	2	2	2	2	2	2	2	3
T	0	1	2	2	3	3	3	3	3	3	3	3
C	0	1	2	2	3	3	4	4	4	4	4	4
G	0	1	2	2	3	3	4	4	5	5	5	5
A	0	1	2	3	3	3	4	5	5	5	5	6

$|LCS| = 6$

Exemplo

$X = \langle GGATCGA \rangle$ e $Y = \langle GAATTCAGTTA \rangle$

		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1	1	1
G	0	1	1	1	1	1	1	1	2	2	2	2
A	0	1	2	2	2	2	2	2	2	2	2	3
T	0	1	2	2	3	3	3	3	3	3	3	3
C	0	1	2	2	3	3	4	4	4	4	4	4
G	0	1	2	2	3	3	4	4	5	5	5	5
A	0	1	2	3	3	3	4	5	5	5	5	6

$|LCS| = 6$

LCS: $\langle GATCGA \rangle$

Exemplo



Exemplo



- $w_1=p_1=7, w_2=p_2=7, w_3=p_3=2, w_4=p_4=3, c=11$

Exemplo

- $w_1=p_1=7, w_2=p_2=7, w_3=p_3=2, w_4=p_4=3, c=11$

B	0	1	2	3	4	5	6	7	8	9	10	11
0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	7	7	7	7	7
2	0	0	0	0	0	0	0	7	7	7	7	7
3	0	0	2	2	2	2	2	7	7	9	9	9
4	0	0	2	3	3	5	5	7	7	9	10	10

Exemplo

- $w_1=p_1=7$, $w_2=p_2=7$, $w_3=p_3=2$, $w_4=p_4=3$, $c=11$

B	0	1	2	3	4	5	6	7	8	9	10	11
0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	7	7	7	7	7
2	0	0	0	0	0	0	0	7	7	7	7	7
3	0	0	2	2	2	2	2	7	7	9	9	9
4	0	0	2	3	3	5	5	7	7	9	10	10

- O preenchimento ótimo da mochila tem valor 10

Exemplo

- $w_1=p_1=7, w_2=p_2=7, w_3=p_3=2, w_4=p_4=3, c=11$

B	0	1	2	3	4	5	6	7	8	9	10	11
0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	7	7	7	7	7
2	0	0	0	0	0	0	0	7	7	7	7	7
3	0	0	2	2	2	2	2	7	7	9	9	9
4	0	0	2	3	3	5	5	7	7	9	10	10

- O preenchimento ótimo da mochila tem valor 10
- Como encontrar os itens dessa solução?

No exemplo anterior (lucro = peso)



No exemplo anterior (lucro = peso)



- Mochila de capacidade 11 e 4 itens

No exemplo anterior (lucro = peso)

- Mochila de capacidade 11 e 4 itens

	w
1	7
2	7
3	2
4	3

No exemplo anterior (lucro = peso)

- Mochila de capacidade 11 e 4 itens

	w	B	0	1	2	3	4	5	6	7	8	9	10	11
1	7	0	0	0	0	0	0	0	0	0	0	0	0	0
2	7	1	0	0	0	0	0	0	0	7	7	7	7	7
3	2	2	0	0	2	2	2	2	2	7	7	9	9	9
4	3	4	0	0	2	3	3	5	5	7	7	9	10	10

No exemplo anterior (lucro = peso)

- Mochila de capacidade 11 e 4 itens
- Preenchimento ótimo tem valor 10

	w	B	0	1	2	3	4	5	6	7	8	9	10	11
0		0	0	0	0	0	0	0	0	0	0	0	0	0
1	7	1	0	0	0	0	0	0	0	7	7	7	7	7
2	7	2	0	0	0	0	0	0	0	7	7	7	7	7
3	2	3	0	0	2	2	2	2	2	7	7	9	9	9
4	3	4	0	0	2	3	3	5	5	7	7	9	10	10

No exemplo anterior (lucro = peso)

- Mochila de capacidade 11 e 4 itens
- Preenchimento ótimo tem valor 10

	w	B	0	1	2	3	4	5	6	7	8	9	10	11	
		0	0	0	0	0	0	0	0	0	0	0	0	0	X
1	7	1	0	0	0	0	0	0	0	7	7	7	7	7	1
2	7	2	0	0	0	0	0	0	0	7	7	7	7	7	2
3	2	3	0	0	2	2	2	2	2	7	7	9	9	9	3
4	3	4	0	0	2	3	3	5	5	7	7	9	10	10	4

No exemplo anterior (lucro = peso)

- Mochila de capacidade 11 e 4 itens
- Preenchimento ótimo tem valor 10

	B	0	1	2	3	4	5	6	7	8	9	10	11	
	0	0	0	0	0	0	0	0	0	0	0	0	0	X
1	7	0	0	0	0	0	0	0	7	7	7	7	7	1
2	7	0	0	0	0	0	0	0	7	7	7	7	7	2
3	2	0	0	2	2	2	2	2	7	7	9	9	9	3
4	3	0	0	2	3	3	5	5	7	7	9	10	10	4

No exemplo anterior (lucro = peso)

- Mochila de capacidade 11 e 4 itens
- Preenchimento ótimo tem valor 10

	B	0	1	2	3	4	5	6	7	8	9	10	11	
	0	0	0	0	0	0	0	0	0	0	0	0	0	X
1	7	0	0	0	0	0	0	0	7	7	7	7	7	1
2	7	0	0	0	0	0	0	0	7	7	7	7	7	2
3	2	0	0	2	2	2	2	2	7	7	9	9	9	3
4	3	0	0	2	3	3	5	5	7	7	9	10	10	4

No exemplo anterior (lucro = peso)

- Mochila de capacidade 11 e 4 itens
- Preenchimento ótimo tem valor 10

	B	0	1	2	3	4	5	6	7	8	9	10	11	
	0	0	0	0	0	0	0	0	0	0	0	0	0	X
1	7	0	0	0	0	0	0	0	7	7	7	7	7	1
2	7	0	0	0	0	0	0	0	7	7	7	7	7	2
3	2	0	0	2	2	2	2	2	7	7	9	9	9	3
4	3	0	0	2	3	3	5	5	7	7	9	10	10	4

No exemplo anterior (lucro = peso)

- Mochila de capacidade 11 e 4 itens
- Preenchimento ótimo tem valor 10

	B	0	1	2	3	4	5	6	7	8	9	10	11	
	0	0	0	0	0	0	0	0	0	0	0	0	0	X
1	7	0	0	0	0	0	0	0	7	7	7	7	7	1
2	7	0	0	0	0	0	0	0	7	7	7	7	7	2
3	2	0	0	2	2	2	2	2	7	7	9	9	9	3
4	3	0	0	2	3	3	5	5	7	7	9	10	10	4

No exemplo anterior (lucro = peso)

- Mochila de capacidade 11 e 4 itens
- Preenchimento ótimo tem valor 10

	B	0	1	2	3	4	5	6	7	8	9	10	11	
	0	0	0	0	0	0	0	0	0	0	0	0	0	X
1	7	0	0	0	0	0	0	0	7	7	7	7	7	1
2	7	0	0	0	0	0	0	0	7	7	7	7	7	2
3	2	0	0	2	2	2	2	2	7	7	9	9	9	3
4	3	0	0	2	3	3	5	5	7	7	9	10	10	4

No exemplo anterior (lucro = peso)

- Mochila de capacidade 11 e 4 itens
- Preenchimento ótimo tem valor 10

	B	0	1	2	3	4	5	6	7	8	9	10	11	
	0	0	0	0	0	0	0	0	0	0	0	0	0	X
1	7	0	0	0	0	0	0	0	7	7	7	7	7	1
2	7	0	0	0	0	0	0	0	7	7	7	7	7	2
3	2	0	0	2	2	2	2	2	7	7	9	9	9	3
4	3	0	0	2	3	3	5	5	7	7	9	10	10	4

No exemplo anterior (lucro = peso)

- Mochila de capacidade 11 e 4 itens
- Preenchimento ótimo tem valor 10

	B	0	1	2	3	4	5	6	7	8	9	10	11	
	0	0	0	0	0	0	0	0	0	0	0	0	0	X
1	7	0	0	0	0	0	0	0	7	7	7	7	7	1
2	7	0	0	0	0	0	0	0	7	7	7	7	7	2
3	2	0	0	2	2	2	2	2	7	7	9	9	9	3
4	3	0	0	2	3	3	5	5	7	7	9	10	10	4

No exemplo anterior (lucro = peso)

- Mochila de capacidade 11 e 4 itens
- Preenchimento ótimo tem valor 10

	B	0	1	2	3	4	5	6	7	8	9	10	11	
	0	0	0	0	0	0	0	0	0	0	0	0	0	X
1	7	0	0	0	0	0	0	0	7	7	7	7	7	1
2	7	0	0	0	0	0	0	0	7	7	7	7	7	0
3	2	0	0	2	2	2	2	2	7	7	9	9	9	0
4	3	0	0	2	3	3	5	5	7	7	9	10	10	1

No exemplo anterior (lucro = peso)

- Mochila de capacidade 11 e 4 itens
- Preenchimento ótimo tem valor 10

	B	0	1	2	3	4	5	6	7	8	9	10	11	
	0	0	0	0	0	0	0	0	0	0	0	0	0	X
1	7	0	0	0	0	0	0	0	7	7	7	7	7	1
2	7	0	0	0	0	0	0	0	7	7	7	7	7	0
3	2	0	0	2	2	2	2	2	7	7	9	9	9	0
4	3	0	0	2	3	3	5	5	7	7	9	10	10	1

- A solução ótima é formada pelos itens 1 e 4, com pesos $w_1=7$ e $w_4=3$.