
Grafos: Busca em Profundidade

SCC0216 Modelagem Computacional em Grafos

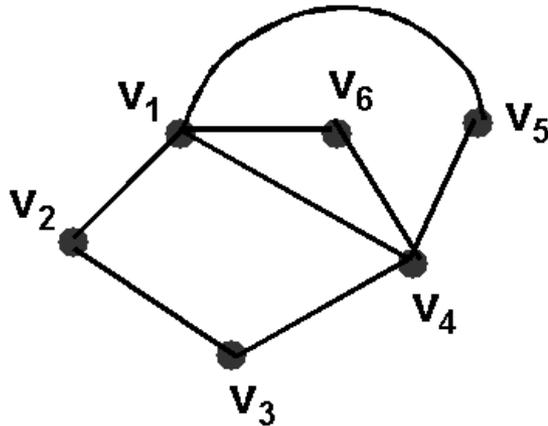
Thiago A. S. Pardo
Maria Cristina F. Oliveira

Componentes Fortemente Conexos (CFC)

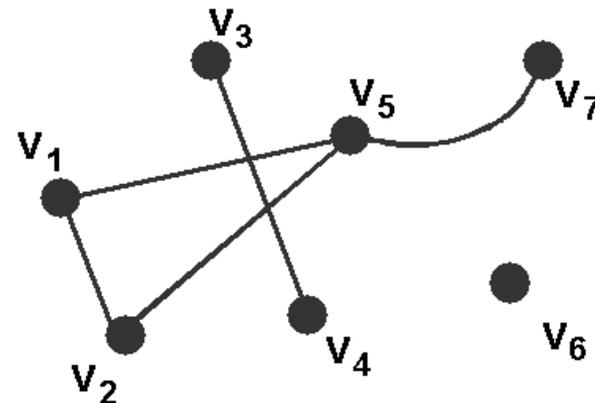
Relembrando (conceitos)

Grafo Conexo (ou conectado)

- Um grafo não orientado $G = (V, A)$ é **conexo** quando existe um caminho entre cada par de vértices de G , caso contrário, G é **desconexo**



Conexo

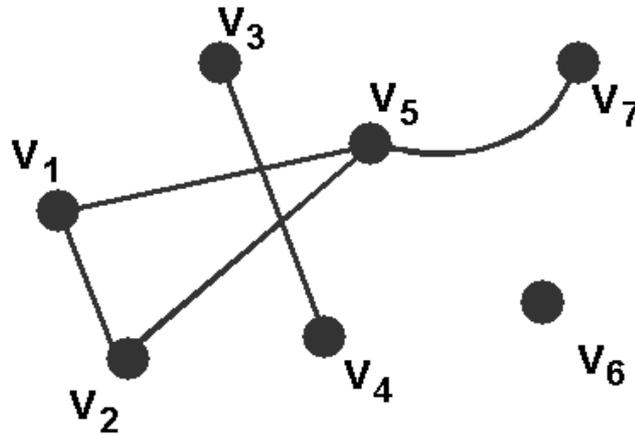


Desconexo

Definições

Componente Conexa

- Uma **componente conexa** corresponde a um **subgrafo conexo**



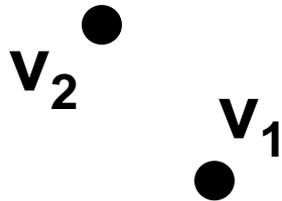
Contém 3 componentes conexas

$\{V_6\}$, $\{V_3, V_4\}$, $\{V_1, V_2, V_5, V_7\}$

Relembrando

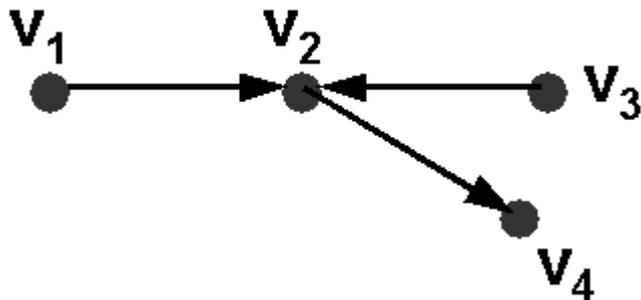
Grafo Conexo

- Um grafo que não possui arestas é **totalmente desconexo**

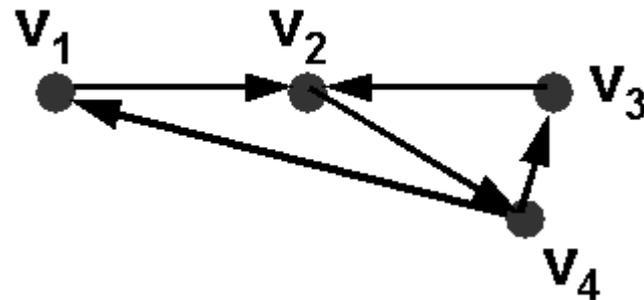


Grafo Orientado Conexo

- Um grafo orientado (dígrafo) $D = (V, A)$ é **conexo** se ao substituir todas as suas arestas por arestas não direcionadas, tem-se um grafo **conexo**, caso contrário, D é **desconexo**



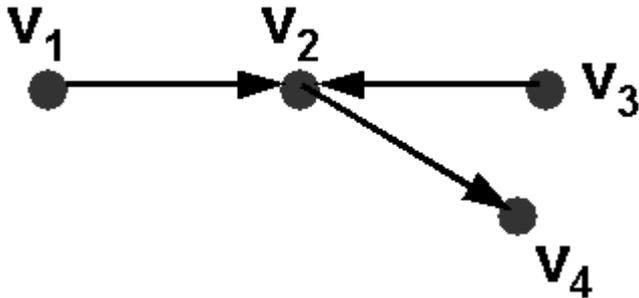
Conexo



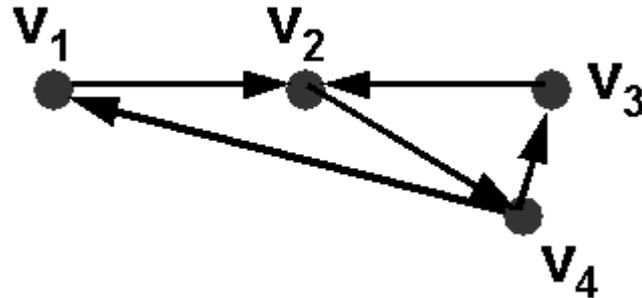
Conexo

Dígrafo Fortemente Conexo

- Um grafo orientado $D = (V,A)$ é dito ser **fortemente conexo** quando existe um caminho entre cada par de vértices (x,y) e também entre (y,x)



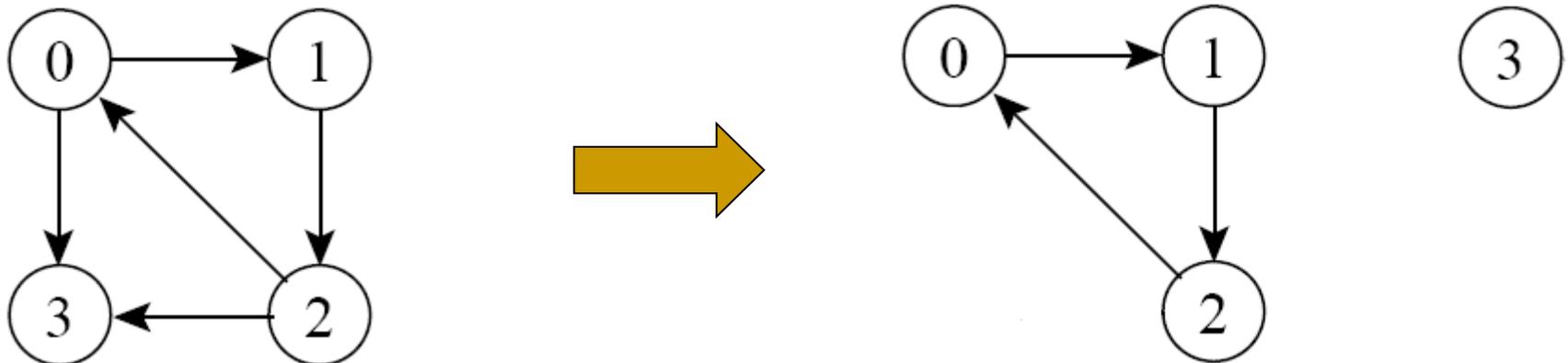
Conexo



Fortemente Conexo

Componentes fortemente conexos

- Um **componente fortemente conexo** de um grafo direcionado é um subconjunto máximo de vértices tal que, para todo par de vértices u e v , u e v são mutuamente alcançáveis



Aplicações

- O conceito de ‘acessibilidade’ é importante em muitos problemas
- Ex. relação entre locais e logística de acesso
 - Grafo modela uma cidade em que os vértices são cruzamentos e arestas são ruas (possivelmente de mão única)
 - Pergunta: como o fechamento de cruzamentos e ruas afeta a acessibilidade a diferentes locais (ex. hospitais, escolas, ...)?

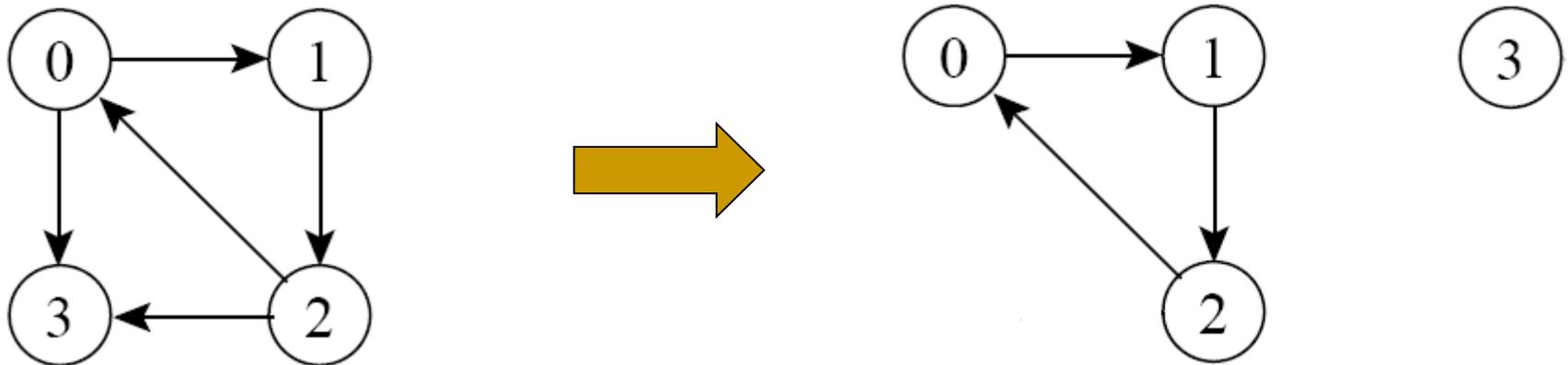
Aplicações

- Como lesões em diferentes regiões do cérebro podem afetar outras regiões?
- Estudo de ‘comunidades’ (grupos de pessoas muito relacionadas) em redes sociais (ou em redes de colaboração)
- Estudo de dependências entre módulos de um software



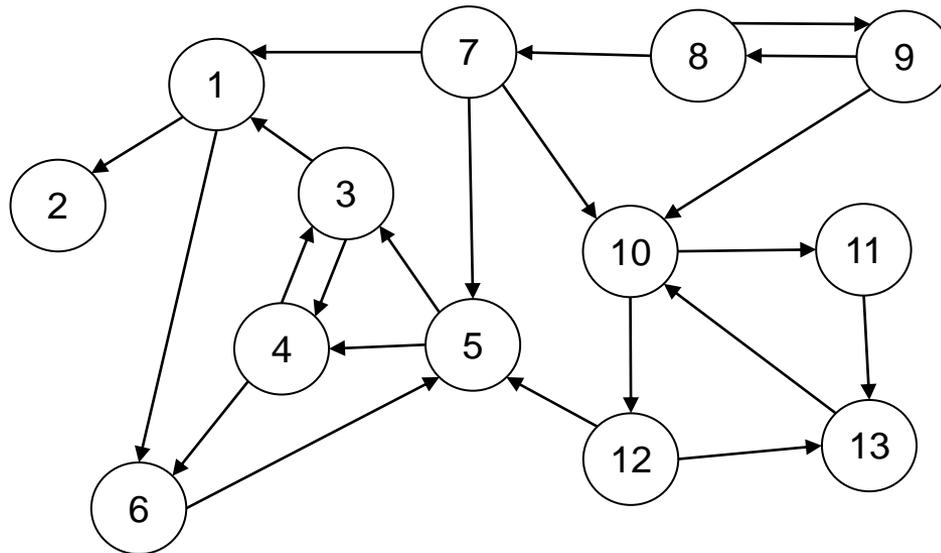
Componentes fortemente conexos

- Um **componente fortemente conexo** de um grafo direcionado é um subconjunto máximo de vértices tal que, para todo par de vértices u e v , u e v são mutuamente alcançáveis



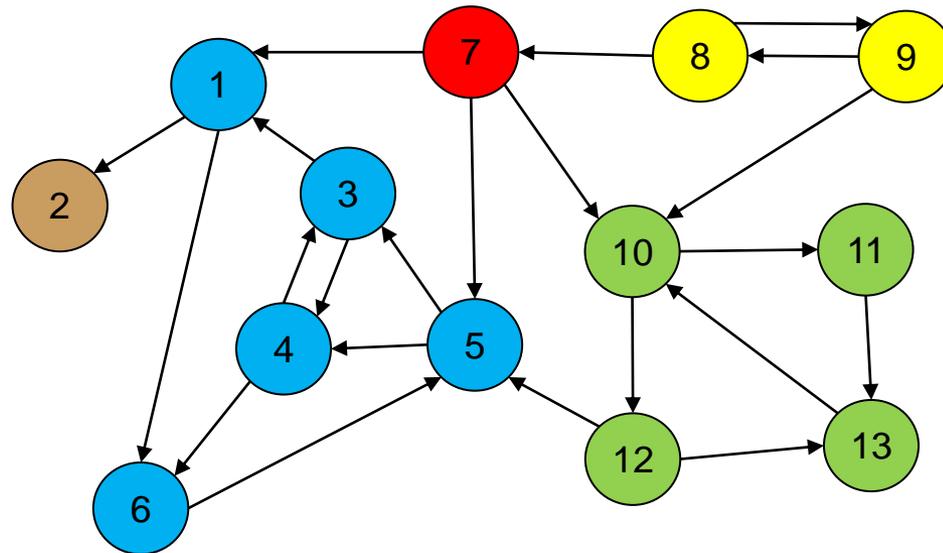
Exercício

- Nem sempre é trivial encontrar esses componentes
 - Quais os componentes fortemente conexos no grafo (dígrafo) abaixo?



Exercício

- Nem sempre é trivial encontrar esses componentes
 - Quais os componentes fortemente conexos no grafo abaixo?

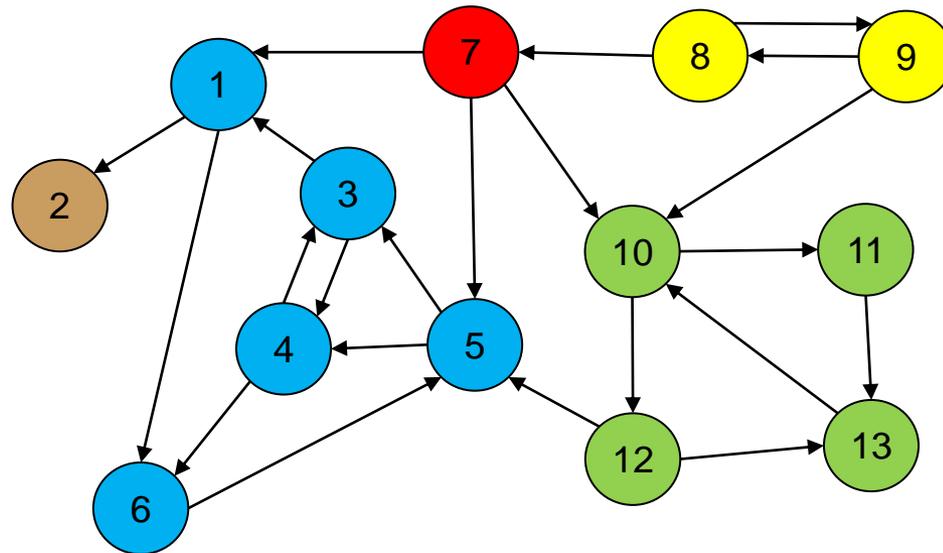


CFC: {2}, {1, 3, 4, 5, 6}, {7}, {8, 9}, {10, 11, 12, 13}

Desafio!

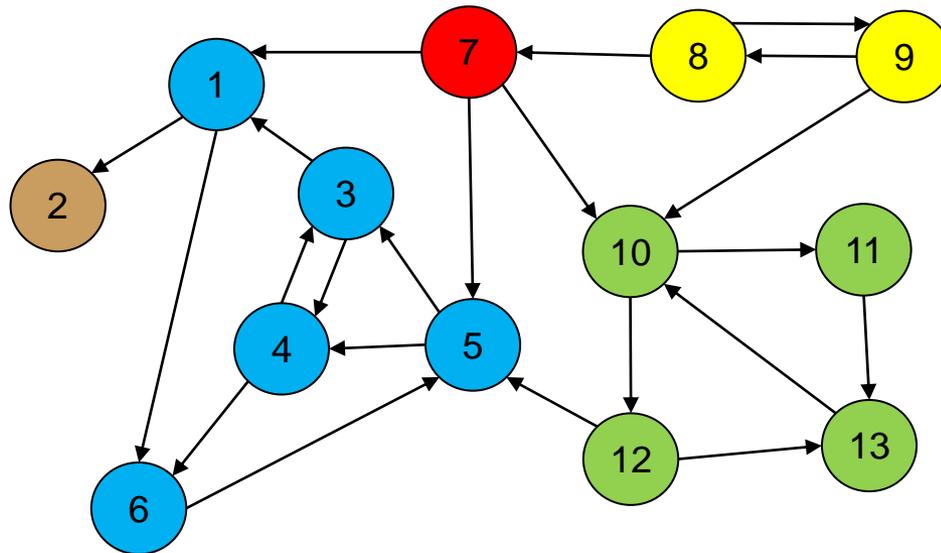
Componentes fortemente conexos

- Propor algoritmo para encontrar componentes fortemente conexos
 - Como fizemos no exemplo abaixo?



Componentes fortemente conexos

- Propor algoritmo para encontrar componentes fortemente conexos
 - Podemos usar a busca em profundidade!

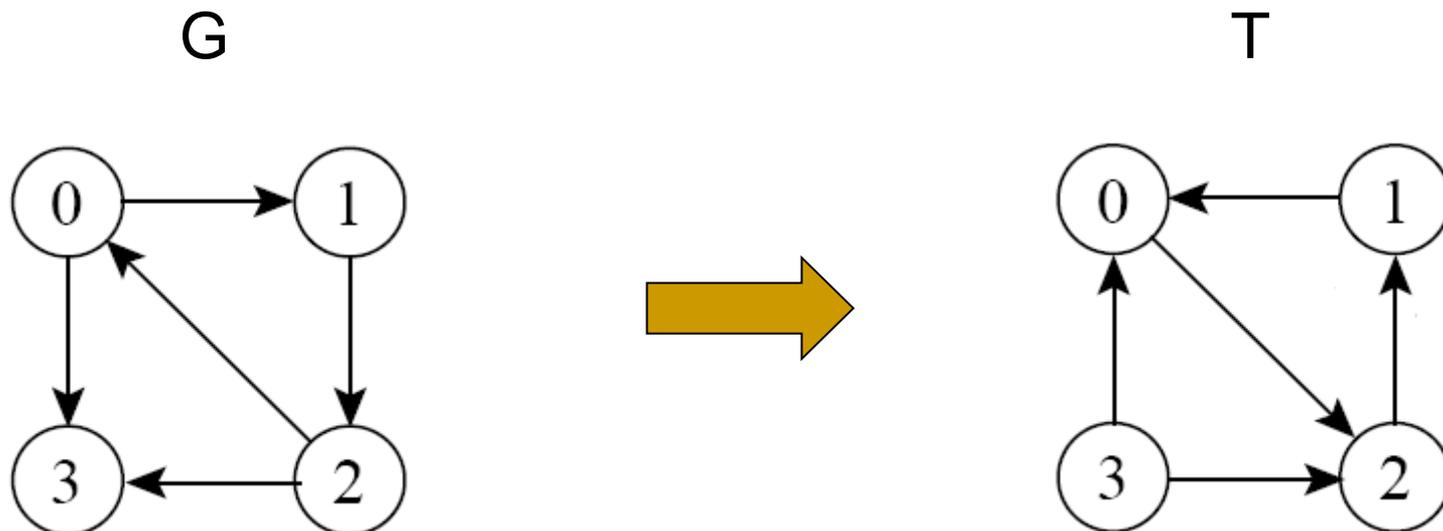


Componentes fortemente conexos

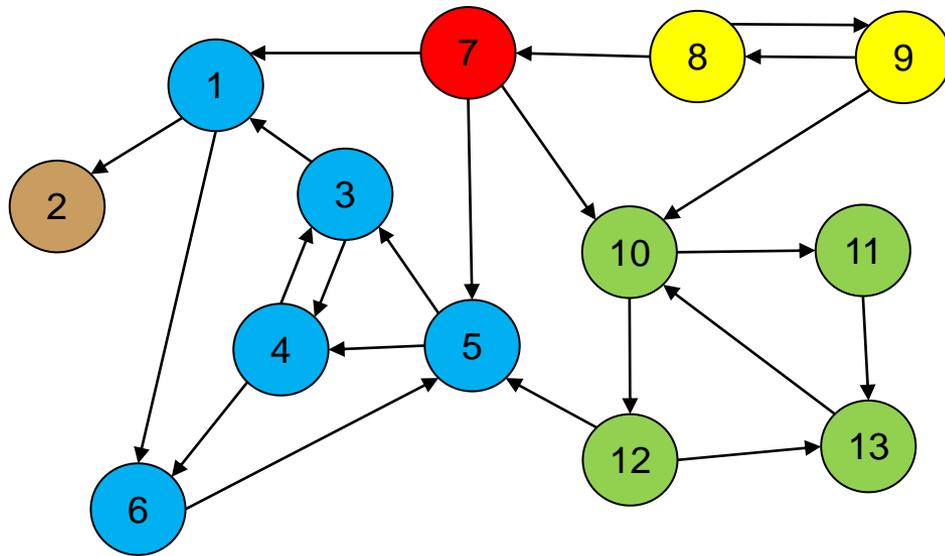
- **Ideia básica** de um algoritmo clássico
 - Um **grafo** direcionado G e sua versão **transposta** T necessariamente possuem os mesmos componentes fortemente conexos
 - Dado $G(V,A)$, o grafo transposto $T(V,A')$
 - Mesmo conjunto V
 - Se existe a aresta (u,v) em G , existe a aresta (v,u) em T

Componentes fortemente conexos

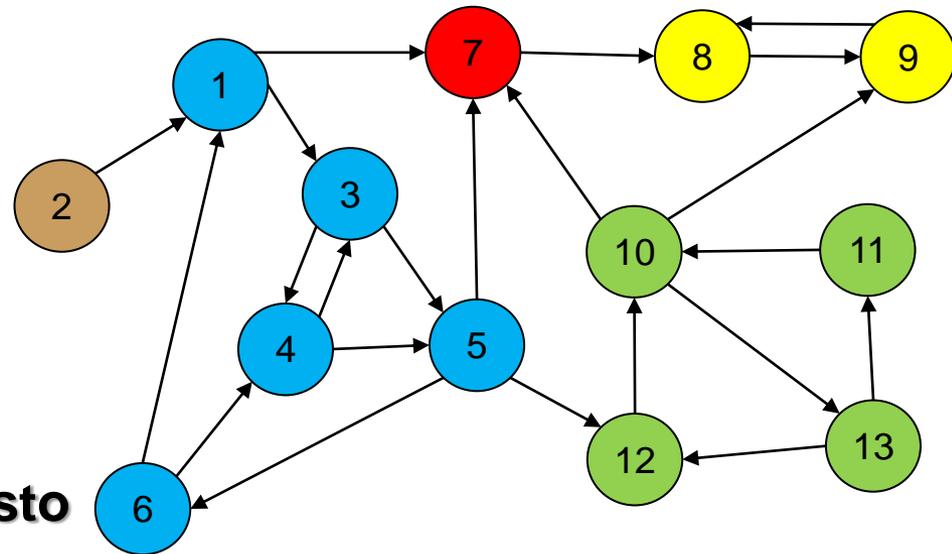
- Exemplo: transpondo um grafo
 - Os componentes fortemente conexos são os mesmos
 - $\{3\}$, $\{0,1,2\}$



Grafo



Grafo Transposto



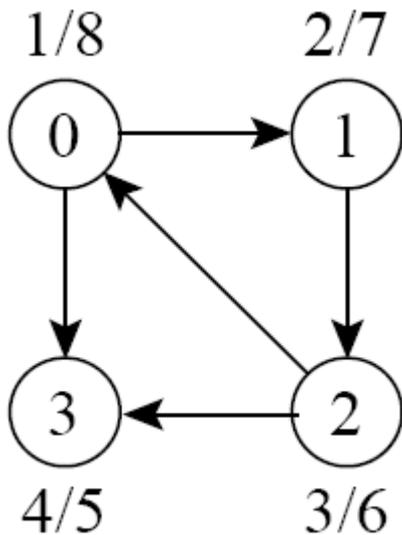
Algoritmo

■ Passo a passo

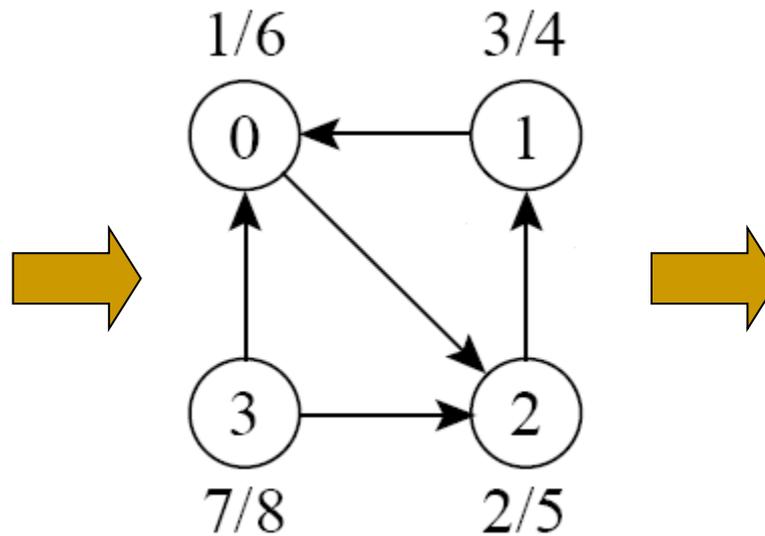
1. Busca em profundidade para determinar tempo de término de cada vértice no grafo original
2. Transposição do grafo
3. Busca em profundidade no grafo transposto **a partir do vértice de maior tempo de término**; se necessário, processo é reiniciado a partir do próximo vértice de maior tempo de término (**ainda não visitado na busca**)
4. Os vértices de cada árvore de busca em profundidade do grafo transposto formam os componentes fortemente conexos

Exemplo

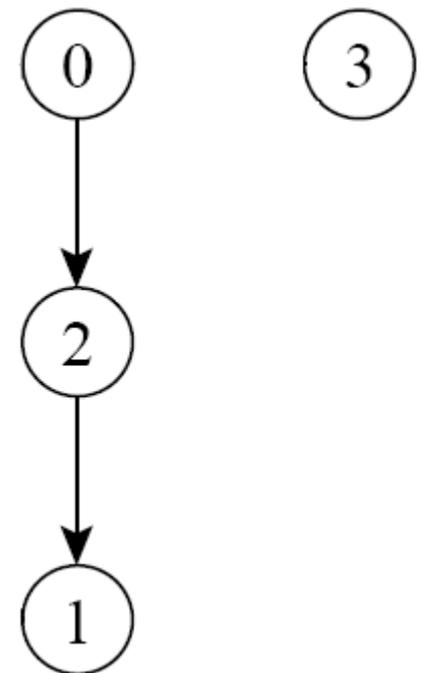
Busca em profundidade



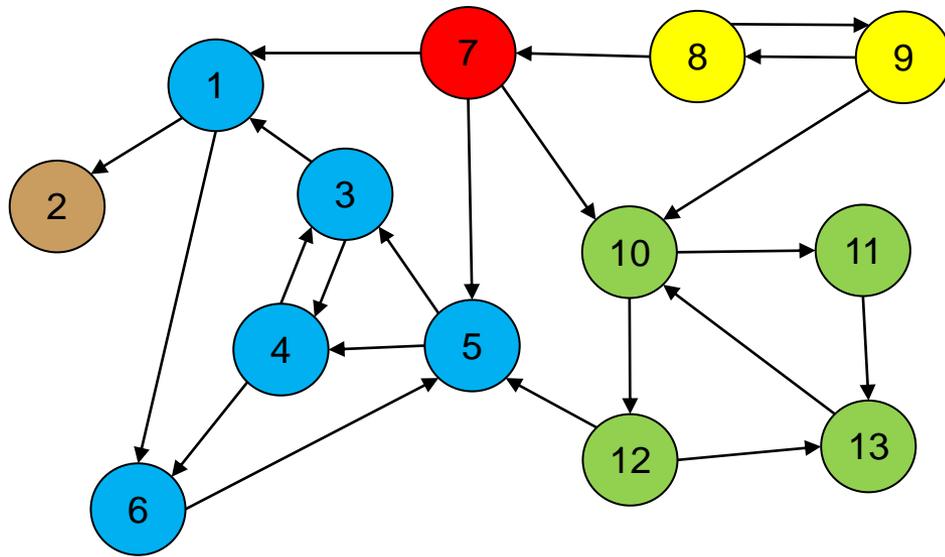
Busca em profundidade no grafo transposto



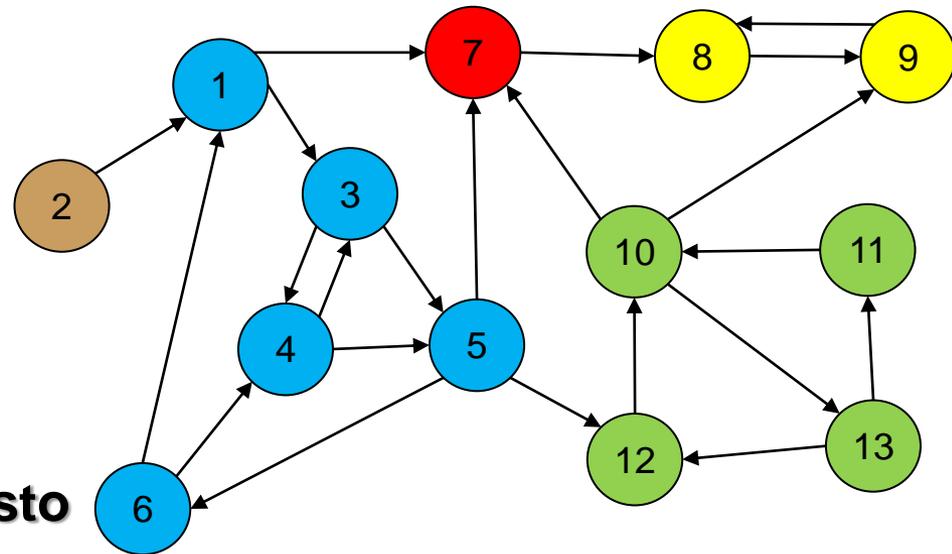
Árvores com componentes



Grafo



Grafo Transposto



Sua opção: 8

*** Sequencia de nos visitados na busca em profundidade ***

No 2, descoberta=2, termino=3, antecessor=1

No 3, descoberta=7, termino=8, antecessor=4

No 4, descoberta=6, termino=9, antecessor=5

No 5, descoberta=5, termino=10, antecessor=6

No 6, descoberta=4, termino=11, antecessor=1

No 1, descoberta=1, termino=12, antecessor=-1

No 13, descoberta=16, termino=17, antecessor=11

No 11, descoberta=15, termino=18, antecessor=10

No 12, descoberta=19, termino=20, antecessor=10

No 10, descoberta=14, termino=21, antecessor=7

No 7, descoberta=13, termino=22, antecessor=-1

No 9, descoberta=24, termino=25, antecessor=8

No 8, descoberta=23, termino=26, antecessor=-1

Pressione qualquer tecla para continuar. . .

Sua opção: 8

*** Sequencia de nos visitados na busca em profundid

No 2, descoberta=2, termino=3, antecessor=1

No 3, descoberta=7, termino=8, antecessor=4

No 4, descoberta=6, termino=9, antecessor=5

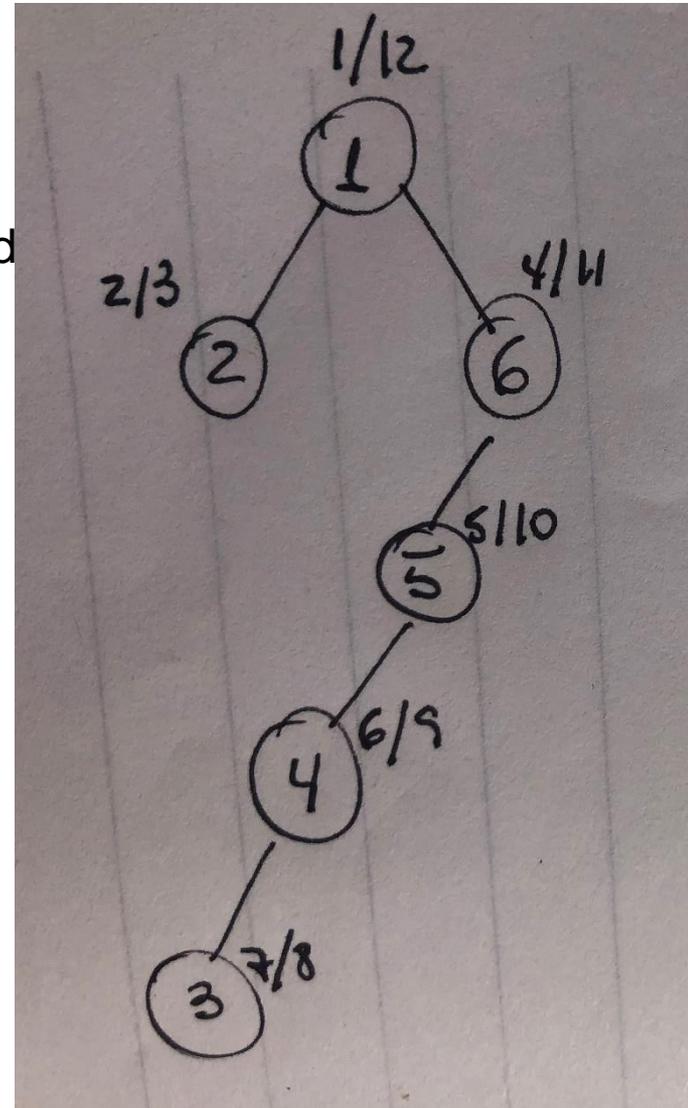
No 5, descoberta=5, termino=10, antecessor=6

No 6, descoberta=4, termino=11, antecessor=1

No 1, descoberta=1, termino=12, antecessor=-1

...

Pressione qualquer tecla para continuar. . .



Sua opção: 8

*** Sequencia de nos visitados na busca em profundidade ***

No 13, descoberta=16, termino=17, antecessor=11

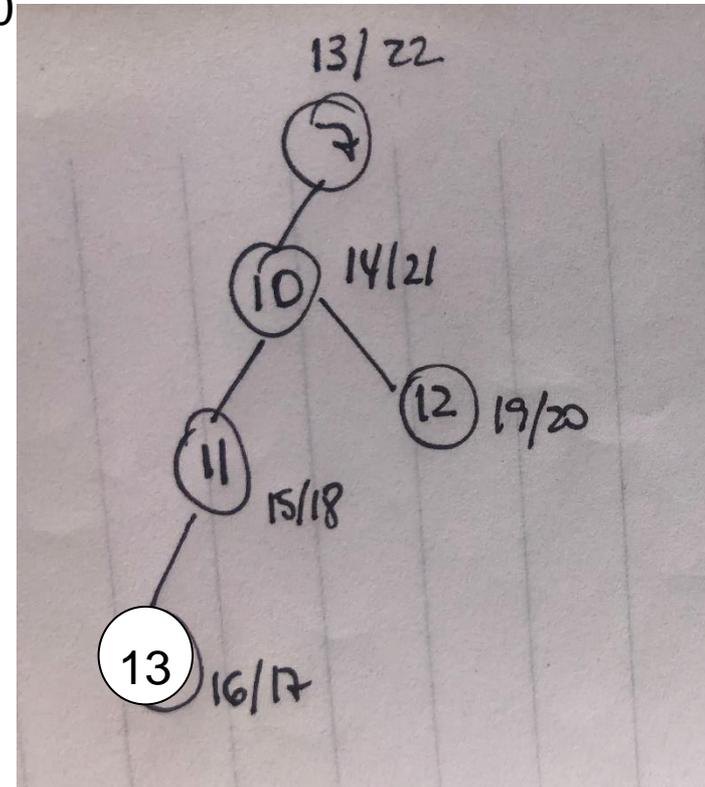
No 11, descoberta=15, termino=18, antecessor=10

No 12, descoberta=19, termino=20, antecessor=10

No 10, descoberta=14, termino=21, antecessor=7

No 7, descoberta=13, termino=22, antecessor=-1

Pressione qualquer tecla para continuar. . .



Sua opção: 8

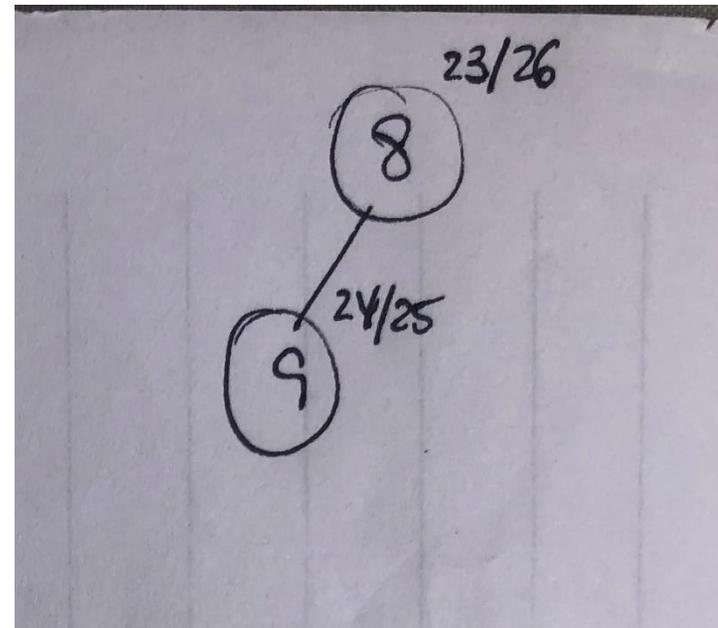
*** Sequencia de nos visitados na busca em profundidade ***

...

No 9, descoberta=24, termino=25, antecessor=8

No 8, descoberta=23, termino=26, antecessor=-1

Pressione qualquer tecla para continuar. . .



Sua opção: 8

*** Sequencia de nos visitados na busca em profundidade ***

No 2, descoberta=2, termino=3, antecessor=1

No 3, descoberta=7, termino=8, antecessor=4

No 4, descoberta=6, termino=9, antecessor=5

No 5, descoberta=5, termino=10, antecessor=6

No 6, descoberta=4, termino=11, antecessor=1

No 1, descoberta=1, termino=12, antecessor=-1

No 13, descoberta=16, termino=17, antecessor=11

No 11, descoberta=15, termino=18, antecessor=10

No 12, descoberta=19, termino=20, antecessor=10

No 10, descoberta=14, termino=21, antecessor=7

No 7, descoberta=13, termino=22, antecessor=-1

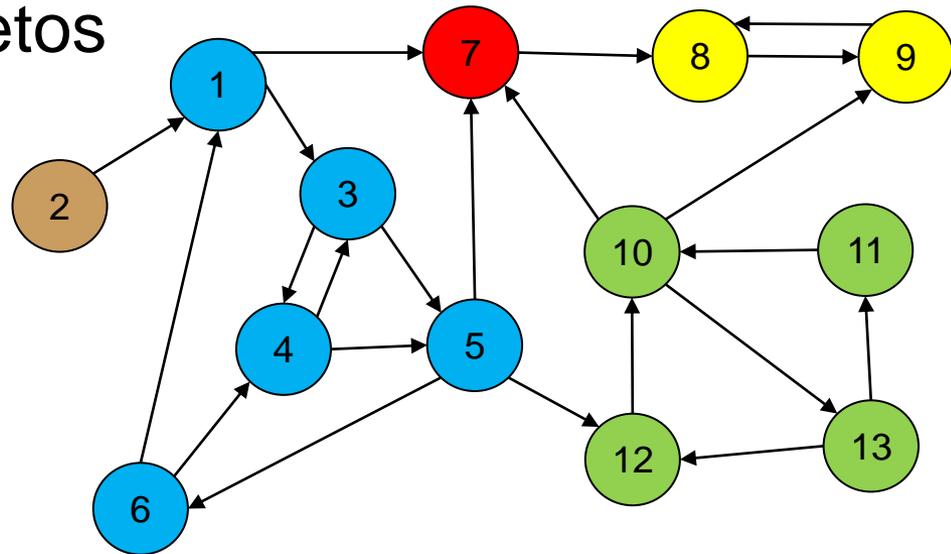
No 9, descoberta=24, termino=25, antecessor=8

No 8, descoberta=23, termino=26, antecessor=-1

Pressione qualquer tecla para continuar. . .

■ Busca DFS no grafo transposto a partir do vértice 8

- Sequencia DFS: 9 8
- vértices 9 e 8 pretos
- $CFC_1 \{8, 9\}$



Sua opção: 8

*** Sequencia de nos visitados na busca em profundidade ***

No 2, descoberta=2, termino=3, antecessor=1

No 3, descoberta=7, termino=8, antecessor=4

No 4, descoberta=6, termino=9, antecessor=5

No 5, descoberta=5, termino=10, antecessor=6

No 6, descoberta=4, termino=11, antecessor=1

No 1, descoberta=1, termino=12, antecessor=-1

No 13, descoberta=16, termino=17, antecessor=11

No 11, descoberta=15, termino=18, antecessor=10

No 12, descoberta=19, termino=20, antecessor=10

No 10, descoberta=14, termino=21, antecessor=7

No 7, descoberta=13, termino=22, antecessor=-1

No 9, descoberta=24, termino=25, antecessor=8

No 8, descoberta=23, termino=26, antecessor=-1

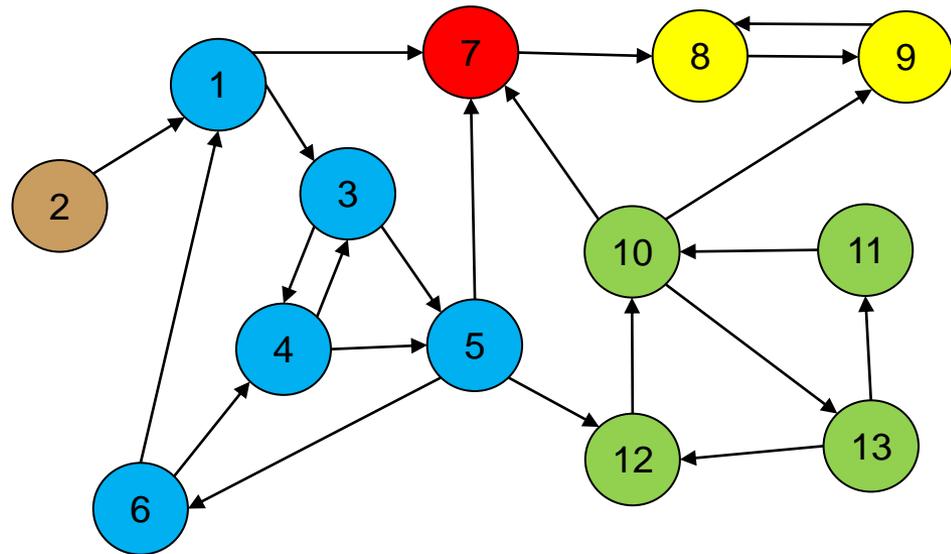
Pressione qualquer tecla para continuar. . .

- Nova busca DFS no grafo transposto a partir do vértice 7

- Sequencia DFS: 7

- 7, 8 e 9 pretos

- $CFC_2 \{7\}$



Sua opção: 8

*** Sequencia de nos visitados na busca em profundidade ***

No 2, descoberta=2, termino=3, antecessor=1

No 3, descoberta=7, termino=8, antecessor=4

No 4, descoberta=6, termino=9, antecessor=5

No 5, descoberta=5, termino=10, antecessor=6

No 6, descoberta=4, termino=11, antecessor=1

No 1, descoberta=1, termino=12, antecessor=-1

No 13, descoberta=16, termino=17, antecessor=11

No 11, descoberta=15, termino=18, antecessor=10

No 12, descoberta=19, termino=20, antecessor=10

No 10, descoberta=14, termino=21, antecessor=7

No 7, descoberta=13, termino=22, antecessor=-1

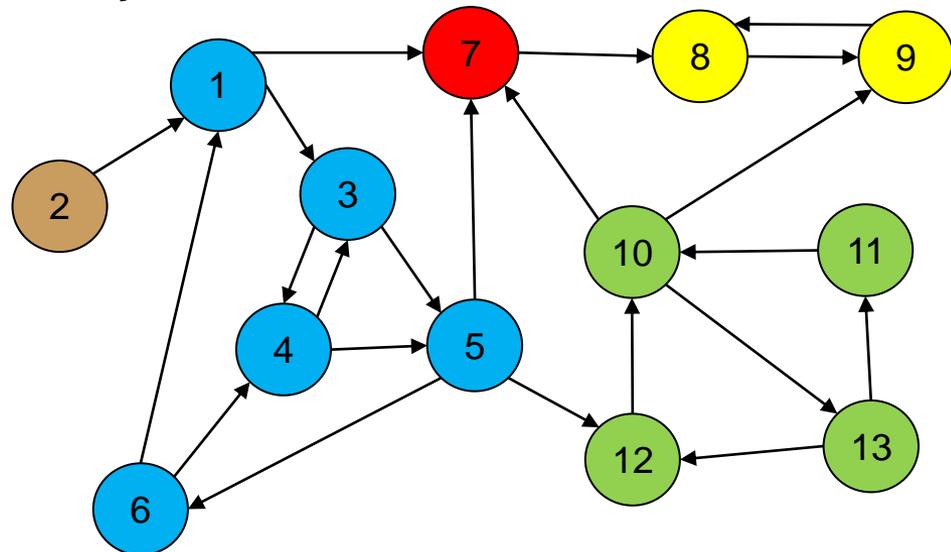
No 9, descoberta=24, termino=25, antecessor=8

No 8, descoberta=23, termino=26, antecessor=-1

Pressione qualquer tecla para continuar. . .

DFS no Grafo: próximo maior tempo de término: vértice 10(21)

- Nova busca DFS no grafo transposto a partir do vértice 10
 - DFS: 11, 12, 13, 10 (8 e 9 já pretos)
 - CFC_3 {10, 11, 12, 13}



Sua opção: 8

*** Sequencia de nos visitados na busca em profundidade ***

No 2, descoberta=2, termino=3, antecessor=1

No 3, descoberta=7, termino=8, antecessor=4

No 4, descoberta=6, termino=9, antecessor=5

No 5, descoberta=5, termino=10, antecessor=6

No 6, descoberta=4, termino=11, antecessor=1

No 1, descoberta=1, termino=12, antecessor=-1

No 13, descoberta=16, termino=17, antecessor=11

No 11, descoberta=15, termino=18, antecessor=10

No 12, descoberta=19, termino=20, antecessor=10

No 10, descoberta=14, termino=21, antecessor=7

No 7, descoberta=13, termino=22, antecessor=-1

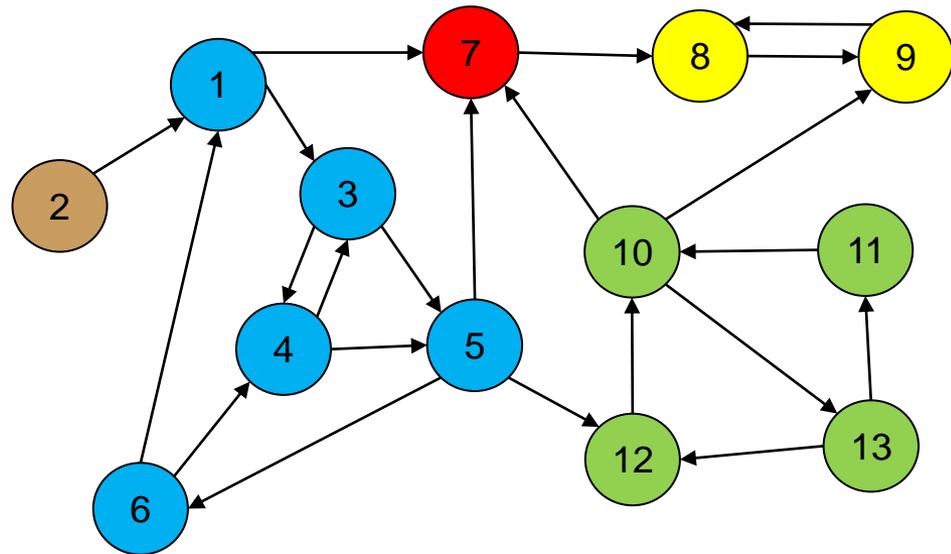
No 9, descoberta=24, termino=25, antecessor=8

No 8, descoberta=23, termino=26, antecessor=-1

Pressione qualquer tecla para continuar. . .

DFS no Grafo: próximo maior tempo de término: vértice 1(12)

- Nova busca DFS no grafo transposto a partir do vértice 1
 - DFS: 6 5 4 3 1 (demais já pretos)
 - $CFC_4 \{1,3,4,5,6\}$



Sua opção: 8

*** Sequencia de nos visitados na busca em profundidade ***

No 2, descoberta=2, termino=3, antecessor=1

No 3, descoberta=7, termino=8, antecessor=4

No 4, descoberta=6, termino=9, antecessor=5

No 5, descoberta=5, termino=10, antecessor=6

No 6, descoberta=4, termino=11, antecessor=1

No 1, descoberta=1, termino=12, antecessor=-1

No 13, descoberta=16, termino=17, antecessor=11

No 11, descoberta=15, termino=18, antecessor=10

No 12, descoberta=19, termino=20, antecessor=10

No 10, descoberta=14, termino=21, antecessor=7

No 7, descoberta=13, termino=22, antecessor=-1

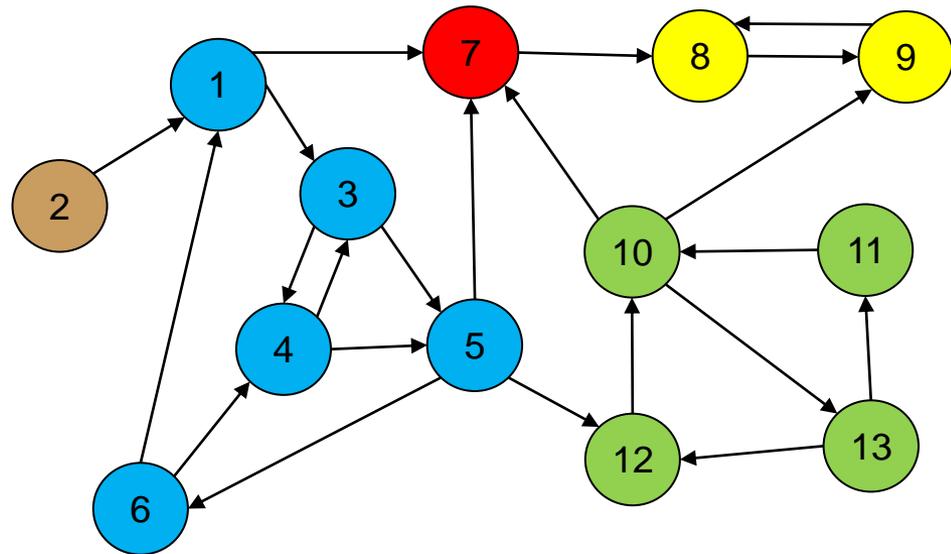
No 9, descoberta=24, termino=25, antecessor=8

No 8, descoberta=23, termino=26, antecessor=-1

Pressione qualquer tecla para continuar. . .

DFS no Grafo: próximo maior tempo de término: vértice 2(3)

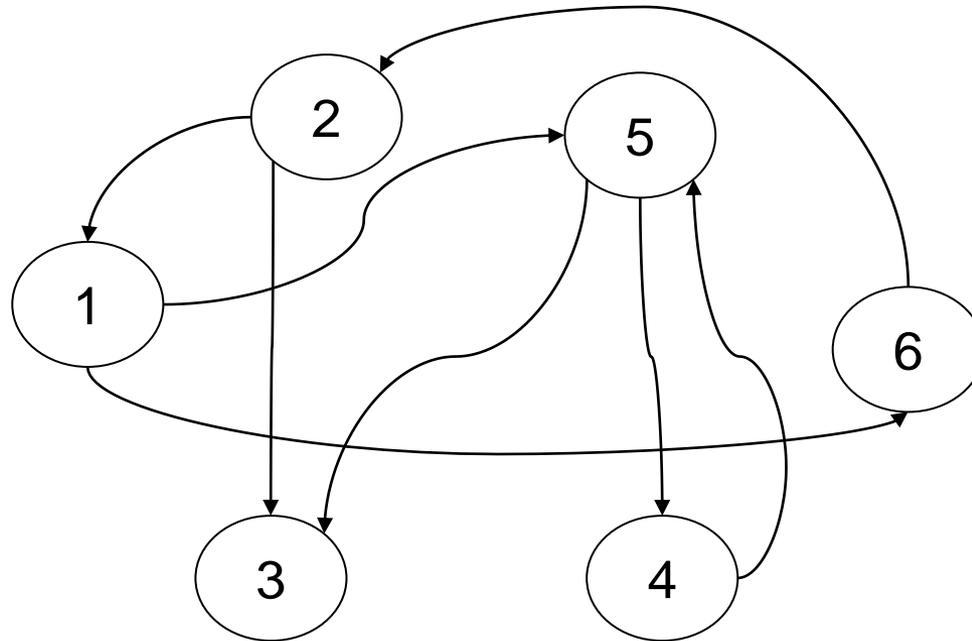
- Nova busca DFS no grafo transposto a partir do vértice 2
 - DFS: 2 (demais já pretos)
 - $CFC_5 \{2\}$



Fim!

Exercício

- Encontre os componentes fortemente conexos do grafo abaixo



Questão

- Qual a complexidade de tempo desse algoritmo?

Algoritmo

■ Passo a passo

1. Busca em profundidade para determinar tempo de término de cada vértice no grafo original
2. Transposição do grafo
3. Busca em profundidade no grafo transposto **a partir do vértice de maior tempo de término**; se necessário, processo é reiniciado a partir do próximo vértice de maior tempo de término (**ainda não visitado na busca**)
4. Os vértices de cada árvore de busca em profundidade do grafo transposto formam os componentes fortemente conexos

Questão

- Qual a complexidade de tempo desse algoritmo?
 - $O(|V| + |A|)$ com representação de listas de adjacências

Componentes fortemente conexos

- Alternativa: algoritmo de Tarjan
 - Proposto por Robert Tarjan em 1972
 - Também faz uso da busca em profundidade
 - Usa uma pilha com “propriedade invariante”
 - Só desempilha em situação específica

DFS: mais uma propriedade

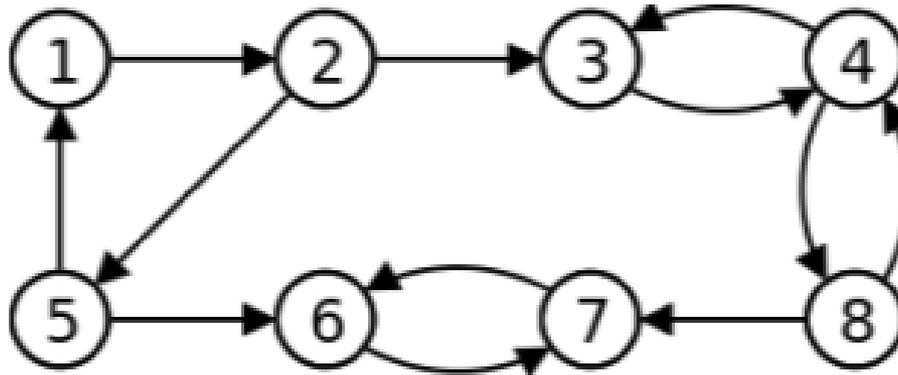
- Possível versão do algoritmo de Tarjan
 - Considera o tempo de descoberta (d) de cada vértice
 - Podemos ignorar o tempo de término
 - No tempo de término (v ficaria preto no DFS) o valor $low[v]$ v será dado pelo menor valor entre
 - $d[v]$ (o tempo de descoberta de v)
 - menor $low(w)$, dados os vértices w conectados a v (i.e., existe aresta (v,w)) por **arestas de árvore**
 - menor $d[w]$, dados os vértices w na Pilha conectados a v por outras arestas

Algoritmo

- Parâmetros de entrada: Grafo G e pilha P inicialmente vazia
- Fazer busca em profundidade e para cada vértice v de G
 - Colocar vértice na pilha P
 - Ao terminar de processar um vértice (que fica preto), calcular e armazenar seu valor low
 - Se $d[v] = low[v]$, então v é a “raiz” de um componente conexo: v e os demais vértices desse componente devem ser retirados da pilha

Exemplo

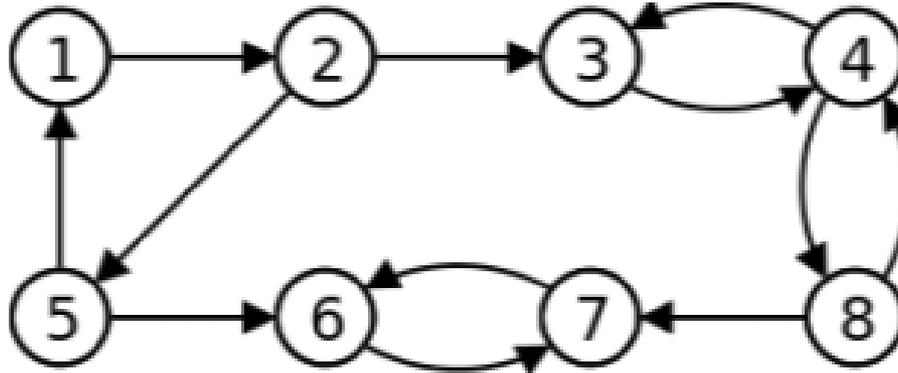
Grafo



CFCs: ?

Exemplo

Grafo

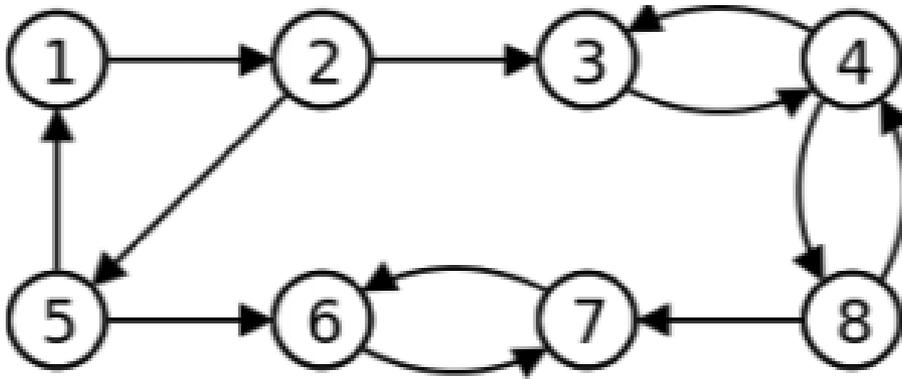


CFCs: {6,7}, {3, 4, 8}, {1, 2, 5}

Exemplo

Árvore de busca
em profundidade,
partindo do vértice 1

Grafo

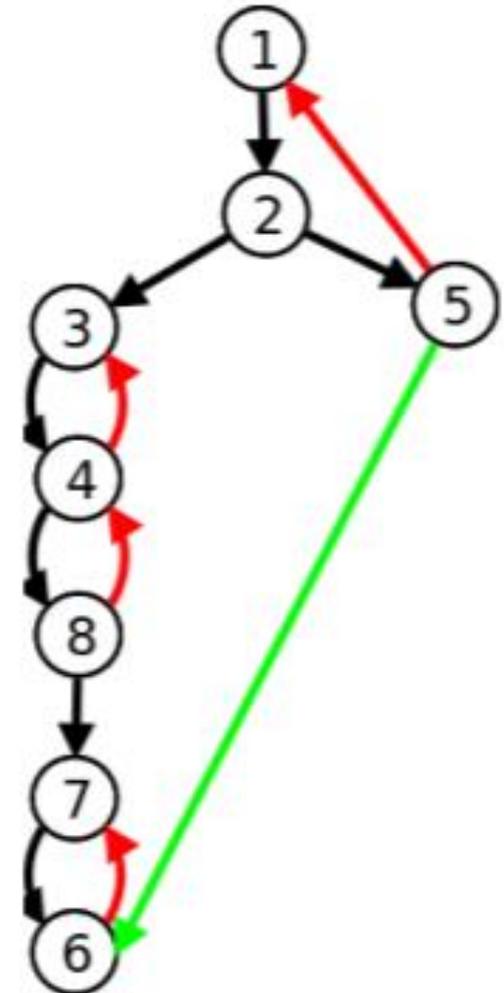
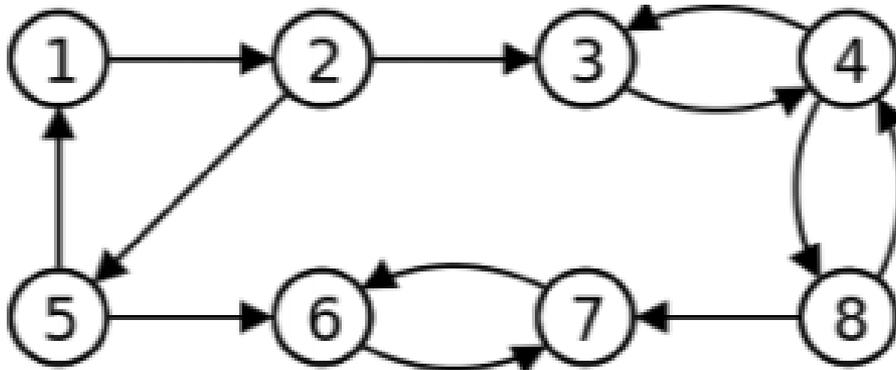


?

Exemplo

Árvore de busca em profundidade

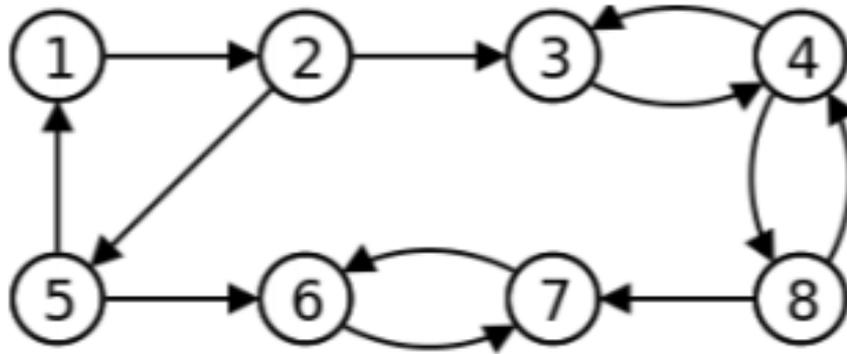
Grafo



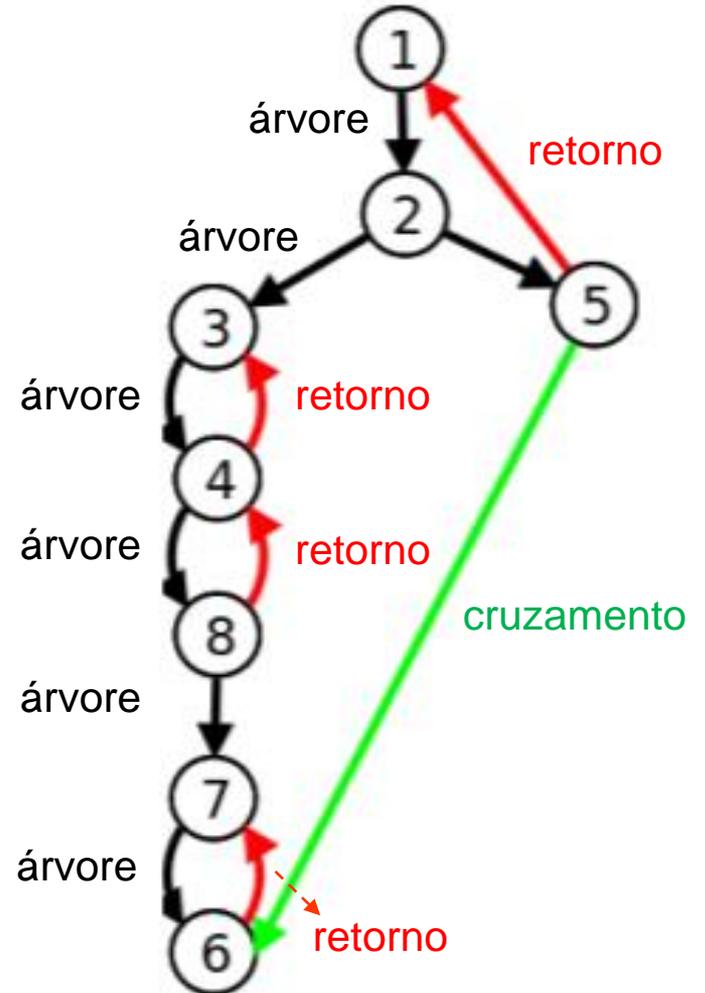
Quais os tipos das arestas?

Exemplo

Grafo



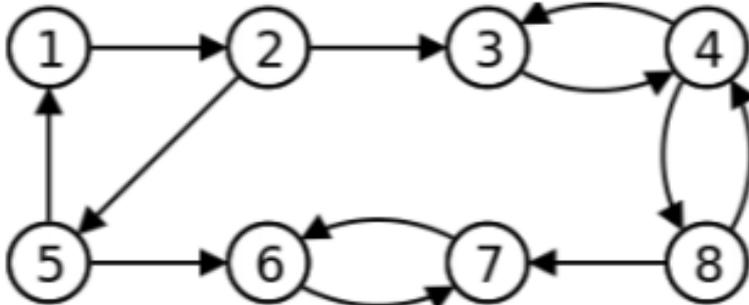
Árvore de busca em profundidade



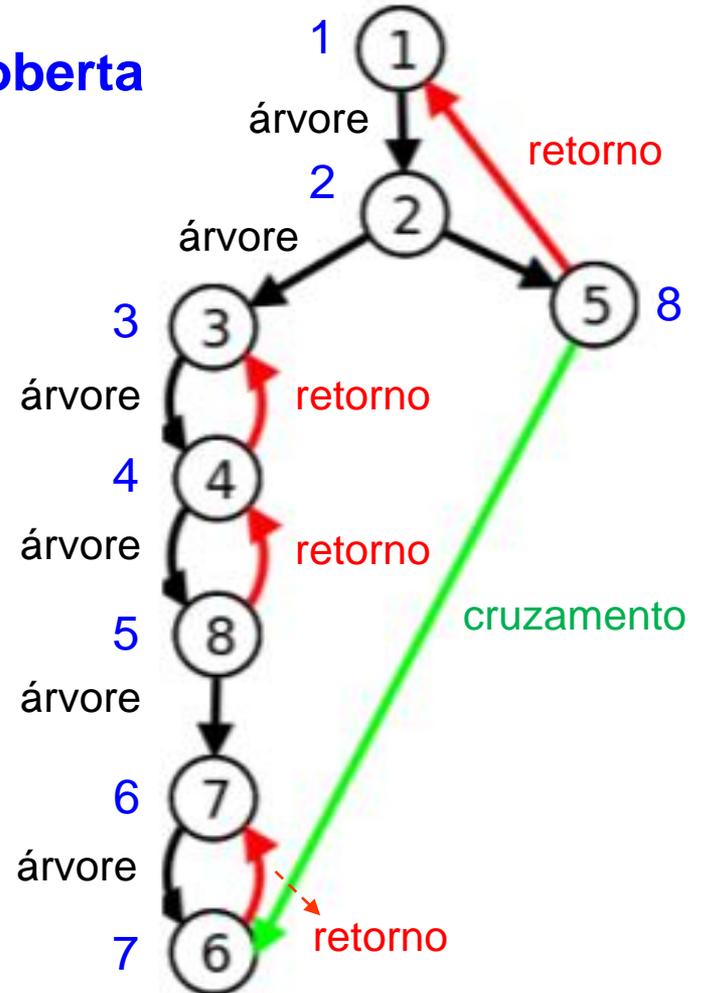
Exemplo

Calculando tempo de descoberta

Grafo



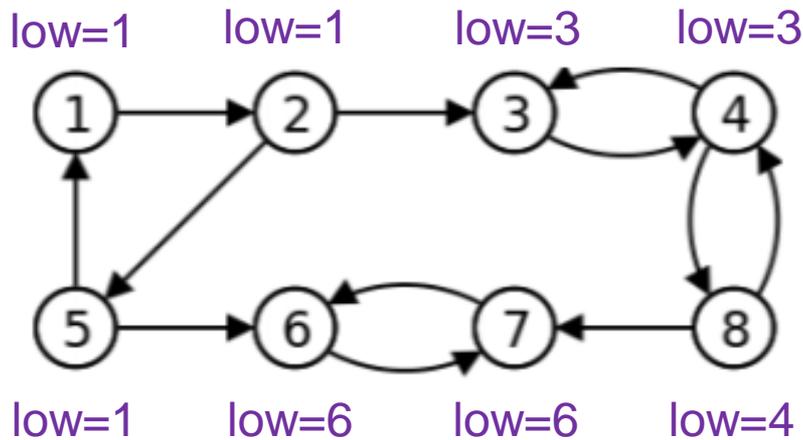
Árvore de busca em profundidade



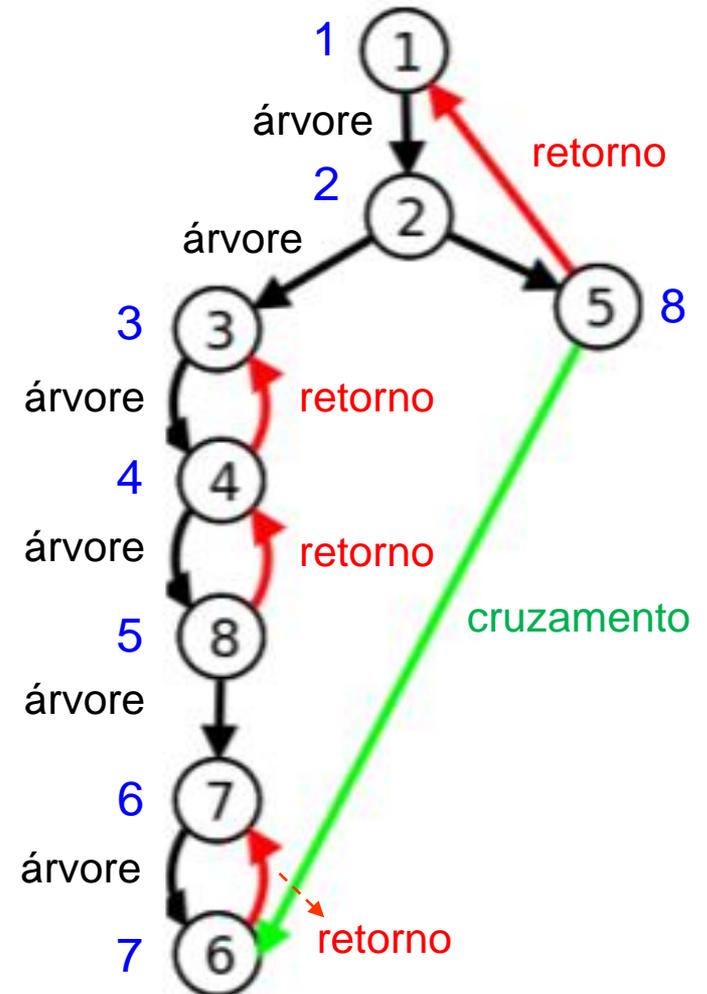
Exemplo

Calculando valor *low* de cada vértice:

Grafo

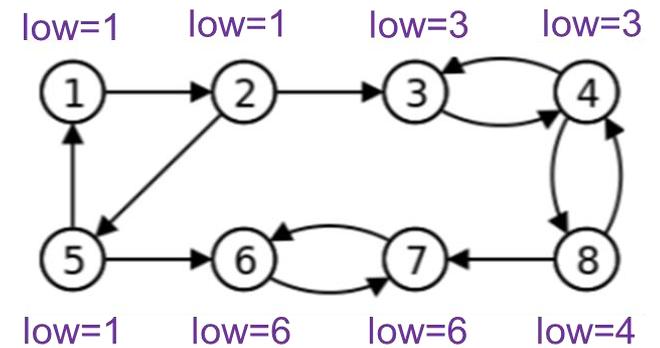
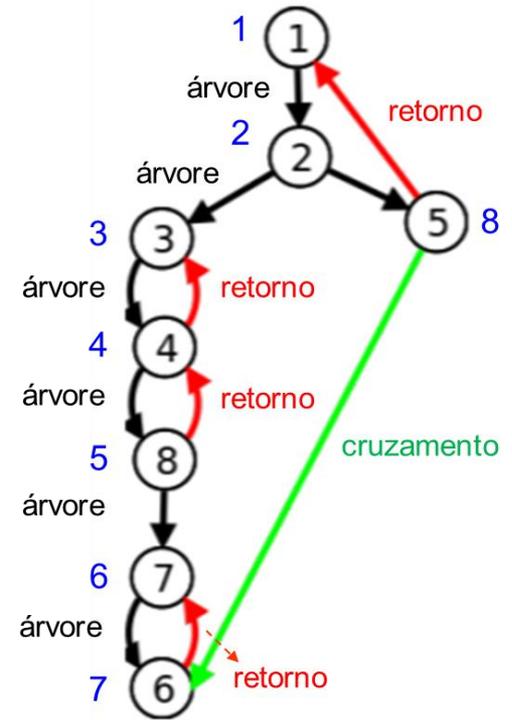


Árvore de busca em profundidade



Exemplo

Pilha P (pilha inicialmente vazia)



Exemplo

Pilha P (empilhando conforme busca)

6 (d=7/low=?)

7 (d=6/low=?)

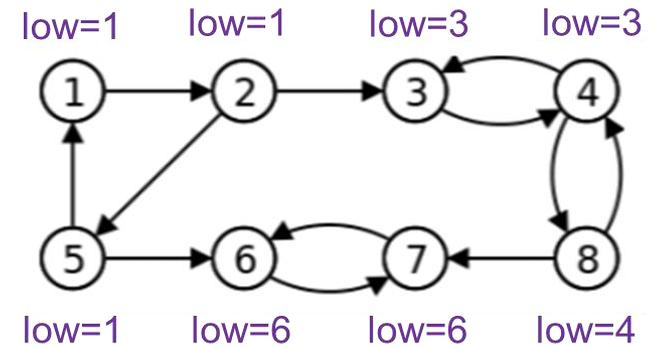
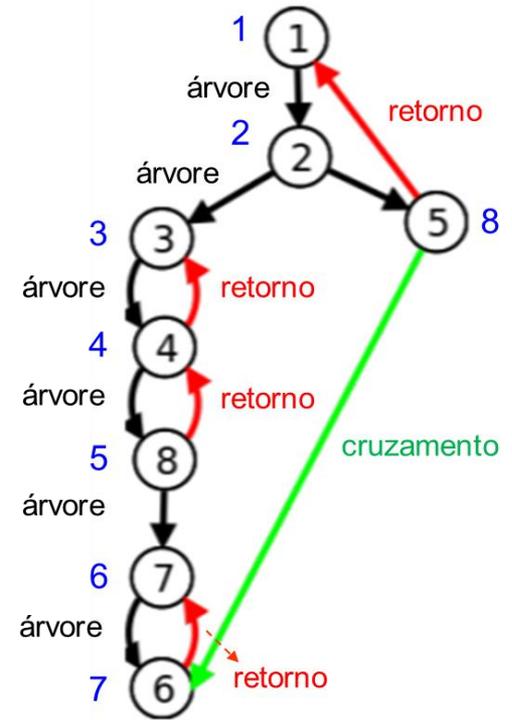
8 (d=5/low=?)

4 (d=4/low=?)

3 (d=3/low=?)

2 (d=2/low=?)

1 (d=1/low=?)



Exemplo

Pilha P (retornando conforme termina)

6 (d=7/low=6)

7 (d=6/low=?)

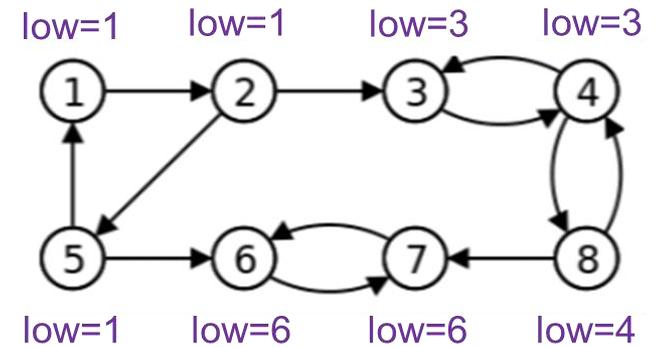
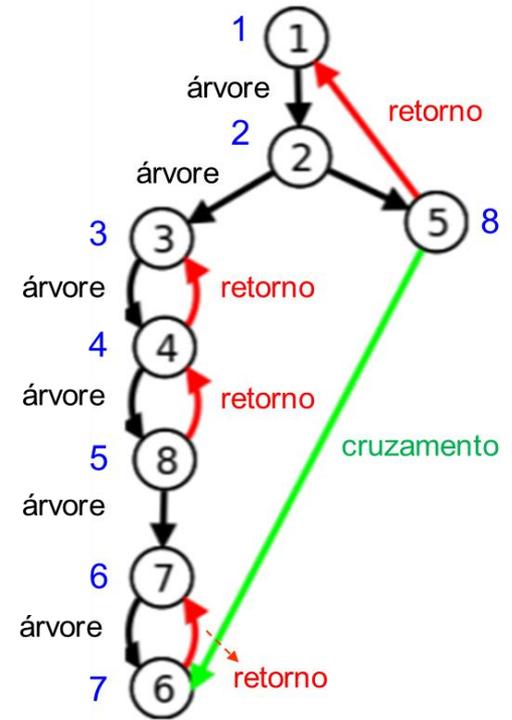
8 (d=5/low=?)

4 (d=4/low=?)

3 (d=3/low=?)

2 (d=2/low=?)

1 (d=1/low=?)



Exemplo

Pilha P (retornando)

6 (d=7/low=6)

7 (d=6/low=6)

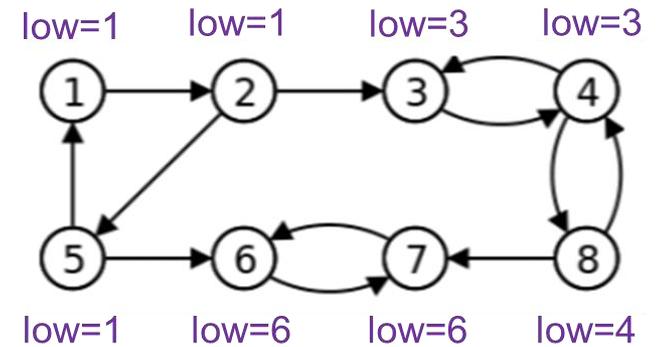
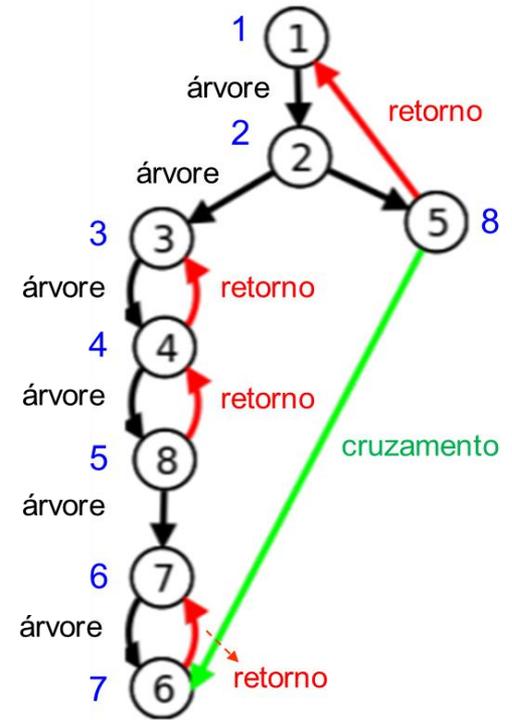
8 (d=5/low=?)

4 (d=4/low=?)

3 (d=3/low=?)

2 (d=2/low=?)

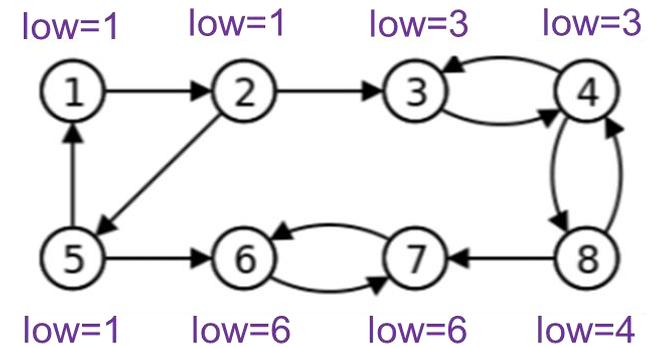
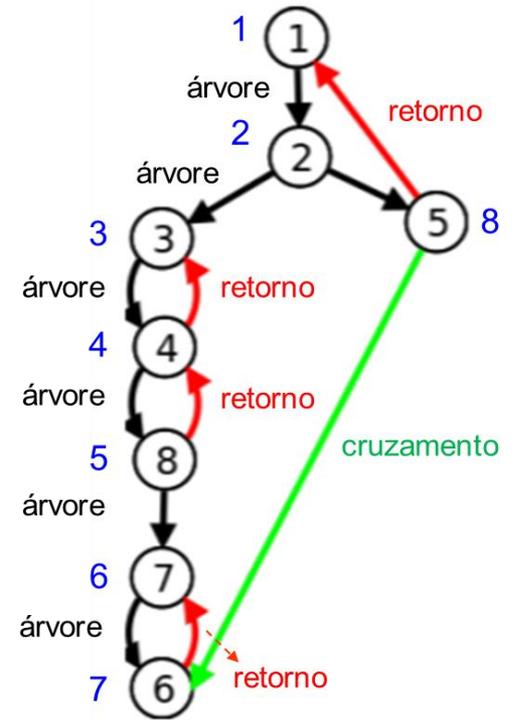
1 (d=1/low=?)



Exemplo

Pilha P (CFC encontrado, desempilha)

6 (d=7/low=6)	}	CFC: 6, 7
7 (d=6/low=6)		
8 (d=5/low=?)		
4 (d=4/low=?)		
3 (d=3/low=?)		
2 (d=2/low=?)		
1 (d=1/low=?)		



Exemplo

Pilha P (retornando)

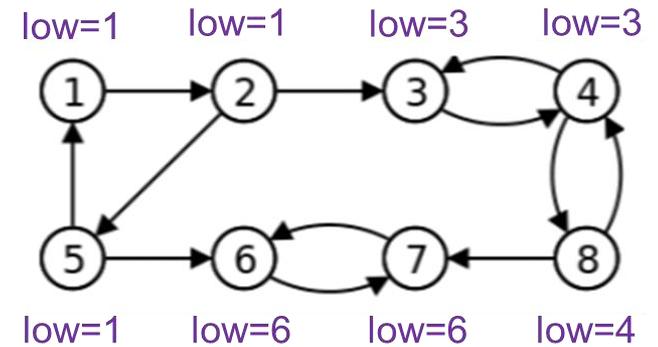
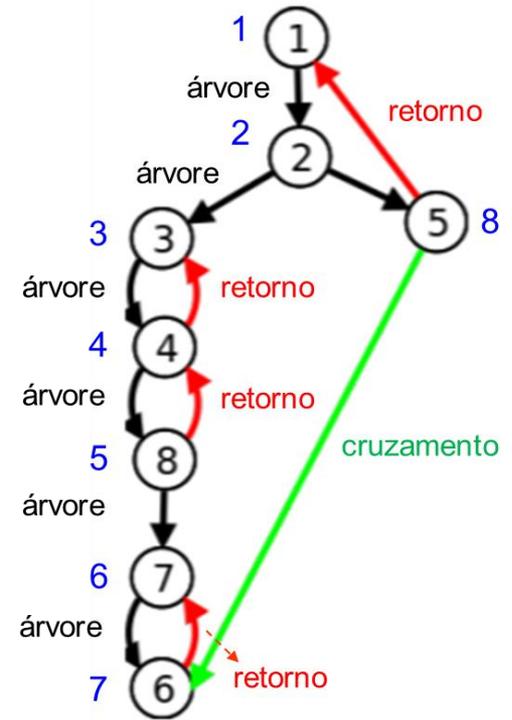
8 (d=5/low=4)

4 (d=4/low=?)

3 (d=3/low=?)

2 (d=2/low=?)

1 (d=1/low=?)



Exemplo

Pilha P (retornando)

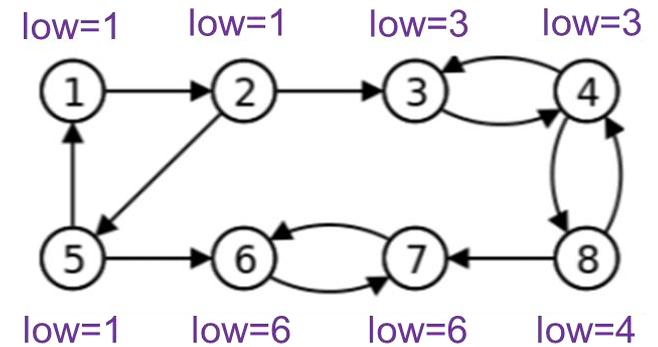
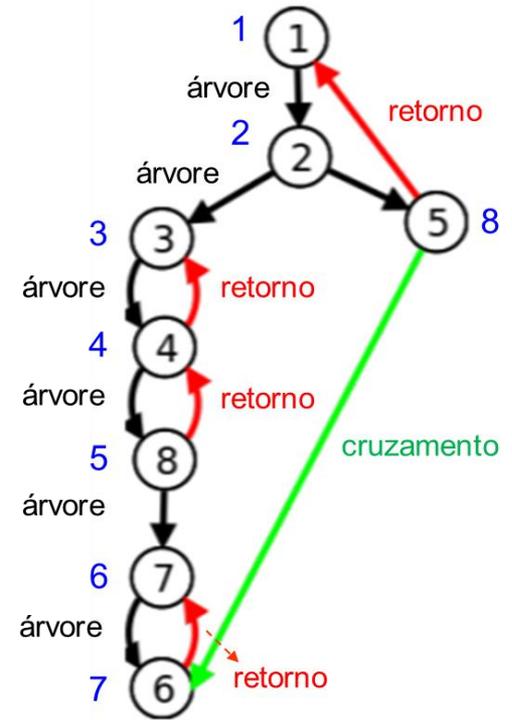
8 (d=5/low=4)

4 (d=4/low=3)

3 (d=3/low=?)

2 (d=2/low=?)

1 (d=1/low=?)



Exemplo

Pilha P (retornando)

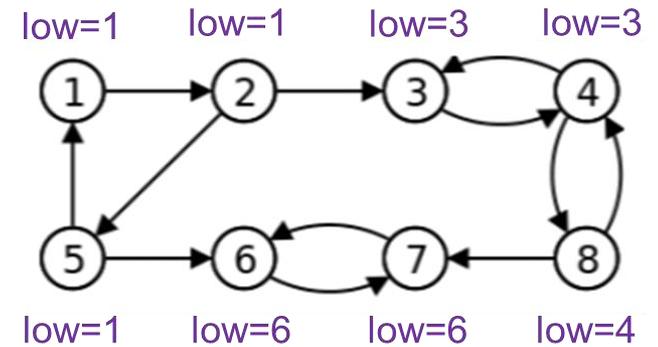
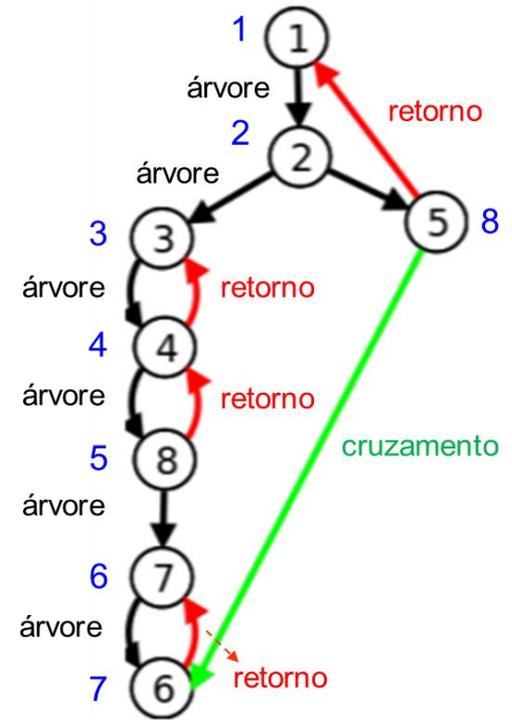
8 (d=5/low=4)

4 (d=4/low=3)

3 (d=3/low=3)

2 (d=2/low=?)

1 (d=1/low=?)

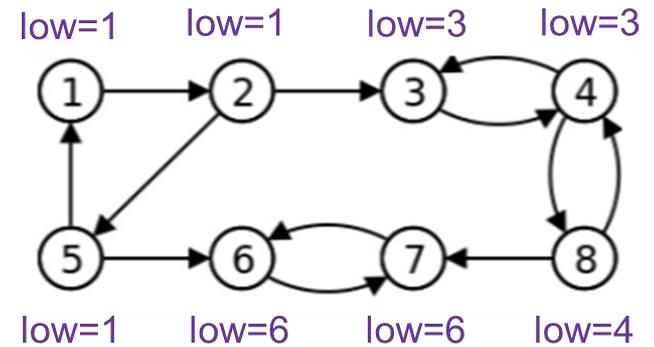
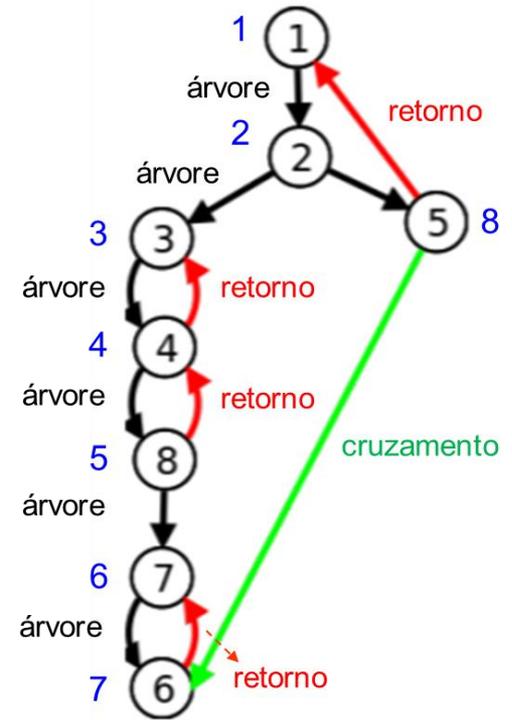


Exemplo

Pilha P (CFC encontrado, desempilha)

8 (d=5/low=4)
 4 (d=4/low=3)
 3 (d=3/low=3)
 2 (d=2/low=?)
 1 (d=1/low=?)

} CFC: 3, 4, 8



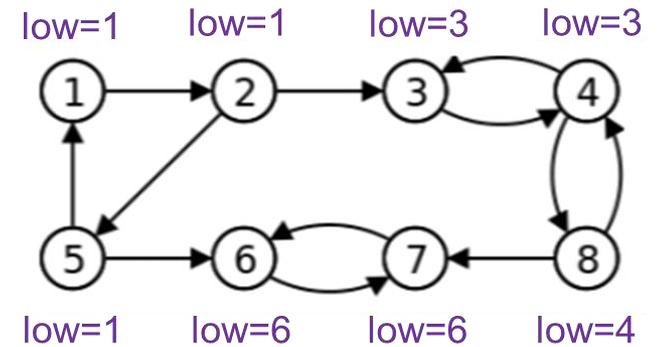
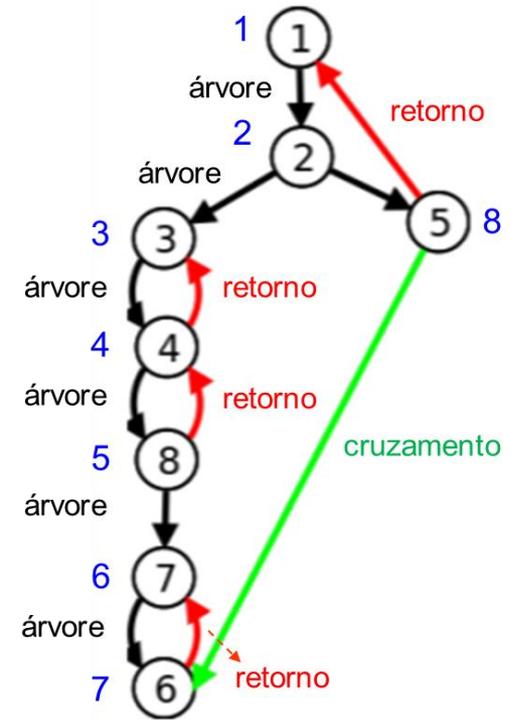
Exemplo

Pilha P (empilhando)

5 ($d=8/\text{low}=?$)

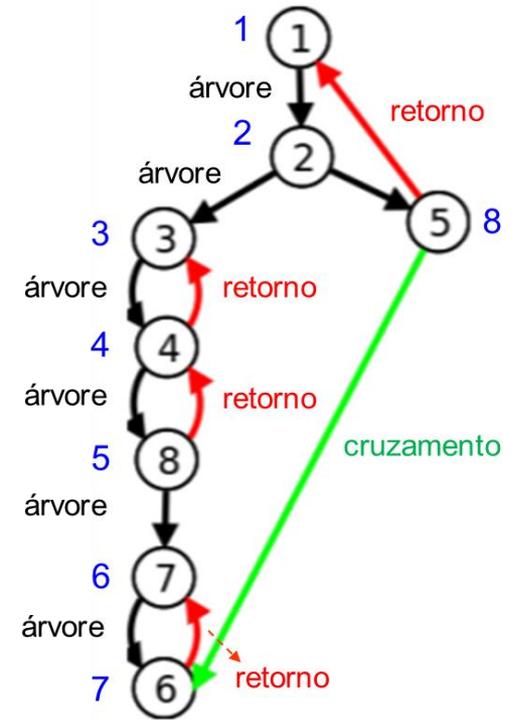
2 ($d=2/\text{low}=?$)

1 ($d=1/\text{low}=?$)



Exemplo

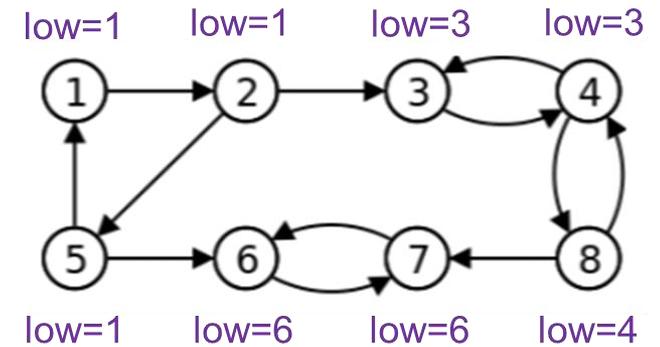
Pilha P (retornando)



5 ($d=8/\text{low}=1$)

2 ($d=2/\text{low}=?$)

1 ($d=1/\text{low}=?$)



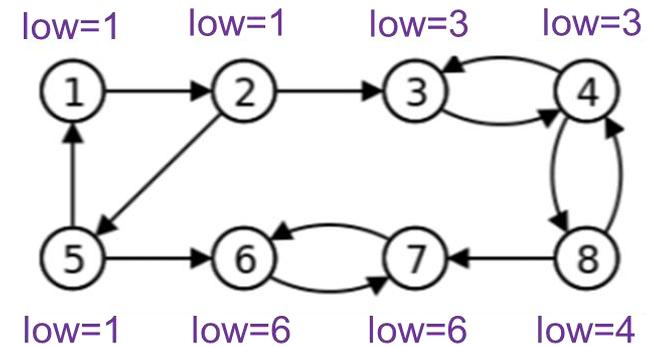
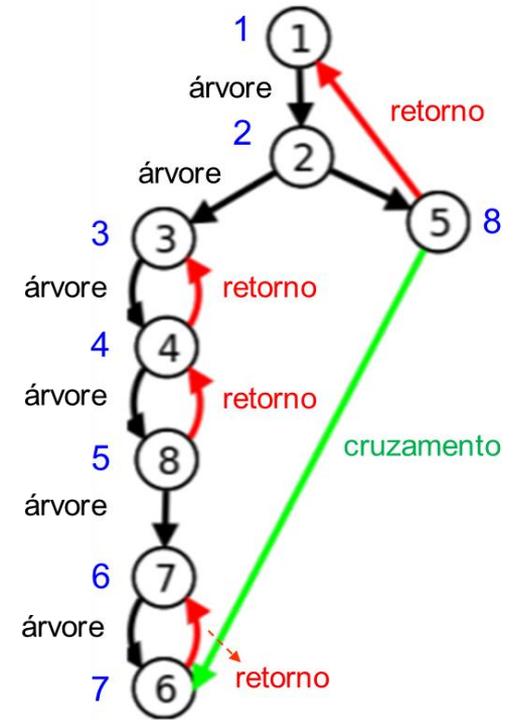
Exemplo

Pilha P (retornando)

5 (d=8/low=1)

2 (d=2/low=1)

1 (d=1/low=?)



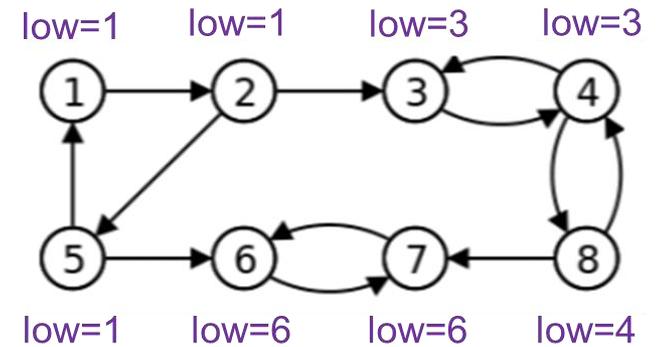
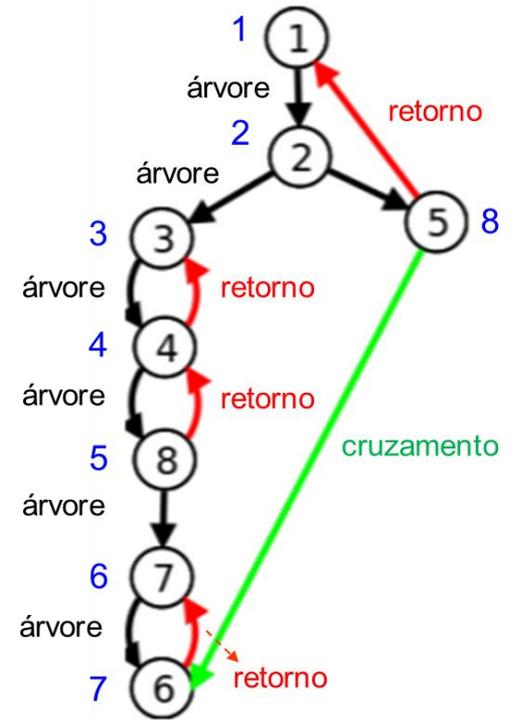
Exemplo

Pilha P (retornando)

5 (d=8/low=1)

2 (d=2/low=1)

1 (d=1/low=1)

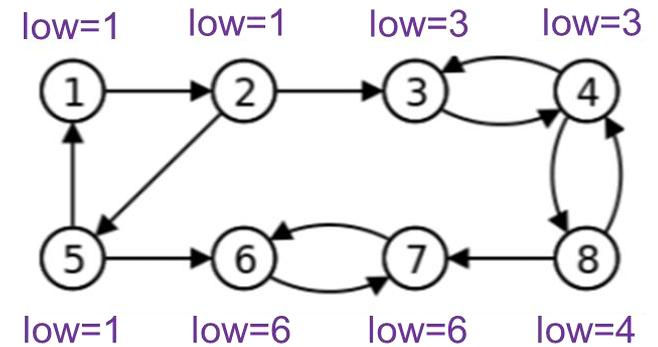
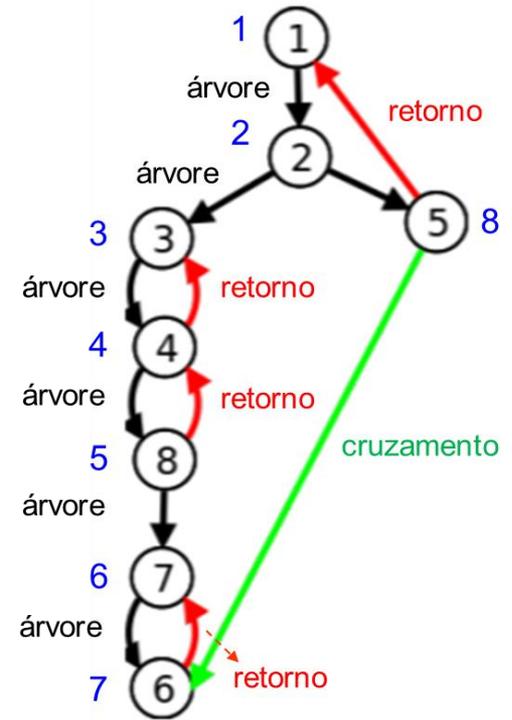


Exemplo

Pilha P (CFC encontrado, desempilha)

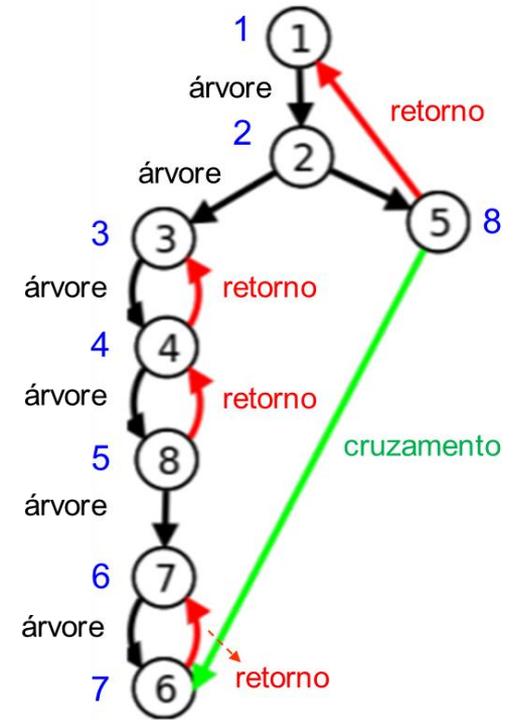
5 (d=8/low=1)
 2 (d=2/low=1)
 1 (d=1/low=1)

} CFC: 1, 2, 5



Exemplo

Pilha P (pilha vazia)

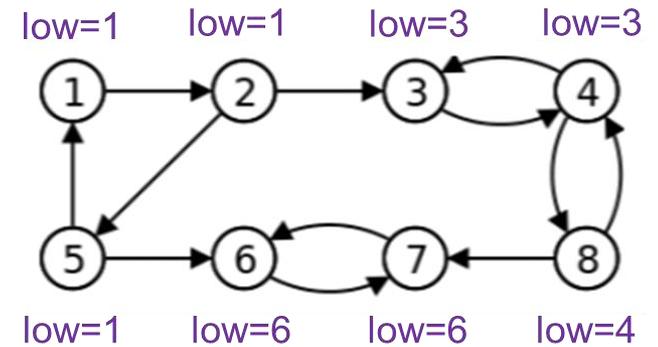


CFCs encontrados

{6, 7}

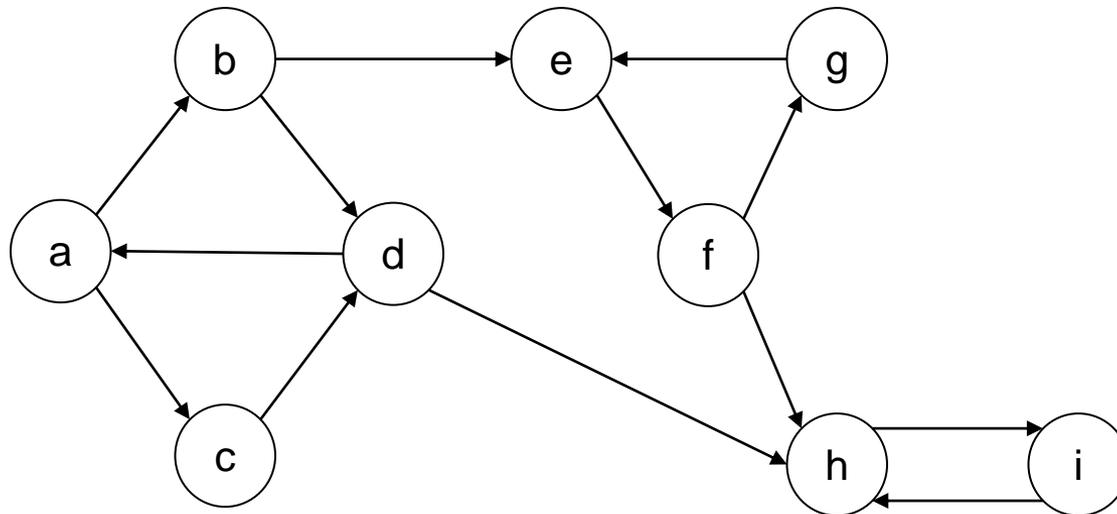
{3, 4, 8}

{1, 2, 5}



Exercício

- Encontrar CFCs no grafo abaixo com o algoritmo de Tarjan



Questão

- Quais as vantagens do algoritmo de Tarjan sobre o clássico?

Questão

- Quais as vantagens do algoritmo de Tarjan sobre o clássico?
 - ❑ Não é necessário transpor o grafo
 - ❑ Basta uma única execução da busca em profundidade (versus 2 execuções no algoritmo clássico)

Algoritmo de Tarjan

- Segundo Donald Knuth (uma das principais referências mundiais em Computação)



<https://cs.stanford.edu/~knuth/>

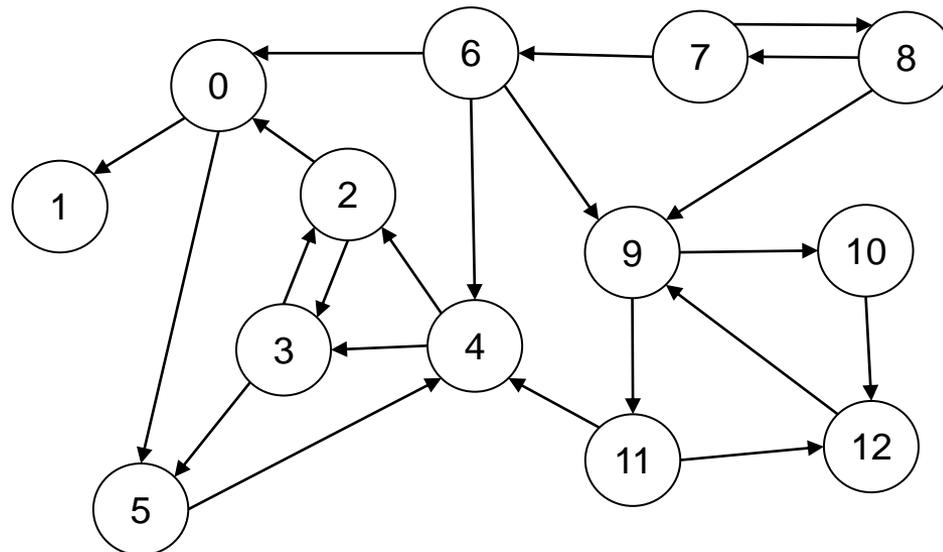
- Uma de suas implementações favoritas
- *The data structures that he [Robert Tarjan] devised for this problem fit together in an amazingly beautiful way, so that the quantities you need to look at while exploring a directed graph are always magically at your fingertips.*

http://www.informit.com/articles/article.aspx?p=2213858&WT.mc_id=Author_Knuth_20Questions

Exercício

Exercício

- Encontrar CFCs do grafo abaixo pelo algoritmo clássico e pelo algoritmo de Tarjan



Exercício

- Implementar algoritmo de Tarjan
- Responder: qual a complexidade de tempo de sua função?