

*UFPA*

# *Algoritmos de Busca em Grafos*

## *Componentes Conectados*

Um grafo não direcionado está conectado se todo par de vértices é conectado por algum percurso.

## *Componentes Conectados*

Um grafo não direcionado está conectado se todo par de vértices é conectado por algum percurso.

### *Relação de Equivalência (Relembrando)*

Uma relação de equivalência é uma relação que possui as propriedades: *Reflexiva*, *Simétrica* e *Transitiva*.

## Componentes Conectados

Um grafo não direcionado está conectado se todo par de vértices é conectado por algum percurso.

### *Relação de Equivalência (Relembrando)*

Uma relação de equivalência é uma relação que possui as propriedades: *Reflexiva*, *Simétrica* e *Transitiva*.

### *Exemplo - Relação de Equivalência:*

Conjunto  $A = \{1, 2, 3, 4\}$  e a relação

$R = \{(1, 1), (1, 2), (2, 1), (2, 2), (3, 4), (4, 3), (3, 3), (4, 4)\}$

## Componentes Conectados

Um grafo não direcionado está conectado se todo par de vértices é conectado por algum percurso.

### *Relação de Equivalência (Relembrando)*

Uma relação de equivalência é uma relação que possui as propriedades: *Reflexiva*, *Simétrica* e *Transitiva*.

### *Exemplo - Relação de Equivalência:*

Conjunto  $A = \{1, 2, 3, 4\}$  e a relação

$R = \{(1, 1), (1, 2), (2, 1), (2, 2), (3, 4), (4, 3), (3, 3), (4, 4)\}$

### *Componentes Conectados*

A relação “está conectado a” é uma de relação de equivalência que divide os vértices em classes de equivalência, i.e. em componentes conectados.

## Componentes Conectados

Um grafo não direcionado está conectado se todo par de vértices é conectado por algum percurso.

### *Relação de Equivalência (Relembrando)*

Uma relação de equivalência é uma relação que possui as propriedades: *Reflexiva*, *Simétrica* e *Transitiva*.

### *Exemplo - Relação de Equivalência:*

Conjunto  $A = \{1, 2, 3, 4\}$  e a relação

$R = \{(1, 1), (1, 2), (2, 1), (2, 2), (3, 4), (4, 3), (3, 3), (4, 4)\}$

### *Componentes Conectados*

A relação “está conectado a” é uma de relação de equivalência que divide os vértices em classes de equivalência, i.e. em componentes conectados.

## *Um algoritmo para encontrar os Componentes Conectados (Sedgewick)*

O algoritmo a seguir permite encontrar as classes de equivalência. Ou seja, ele determina quais vértices estão conectados. Para isso, vamos utilizar o DFS.

## *Um algoritmo para encontrar os Componentes Conectados (Sedgewick)*

O algoritmo a seguir permite encontrar as classes de equivalência. Ou seja, ele determina quais vértices estão conectados. Para isso, vamos utilizar o DFS.

---

```
public class ComponentesConectados {  
    private boolean[] marcado;  
    private int[] id;  
    private int cont;  
    public ComponentesConectados(Grafo G) {  
        marcado = new boolean[G.V()];  
        id = new int[G.V()]  
        for (int s = 0; s < G.V(); s++)  
            if ( !marcado[s]) {  
                dfs(G,s);  
                cont++;  
            }  
    }  
}
```

---



## *Um algoritmo para encontrar os Componentes Conectados (cont.)*

$id[v] = i$  se o vertice ' $v$ ' está no  $i$ 'th componente conectado processado. Ele Associa o mesmo valor para todos os vértices no mesmo componente.

---

```
private void dfs(Grafo G, int v) {  
    marcado[v] = true;  
    id[v] = cont;  
    for (int w: G.adj())  
        if (!marcado[w])  
            dfs(G,w);  
}
```

---

Para cada vértice  $v$  no 1o. componente  $id[v] = 0$ ; para cada vértice  $v$  no 2o. componente  $id[v] = 1$ ; etc.

## *Um algoritmo para encontrar os Componentes Conectados (cont.)*

---

```
// Verifica se dois vertices estao conectados
public boolean conectado(int v, int w)
{ return id[v] == id[w]; }
//Identificador do componente para v
public int id(int v) {
    return id[v];
}
// Numero de componentes conectados
public int cont() { return cont; }
}
```

---

## Componentes Conectados - Exemplo/Exercício

Utilize o algoritmo para encontrar os componentes fortemente conectados nos seguintes grafos:

*Exemplo 1:*

$G(V, E)$  onde  $V = \{1, 2, 3, 4, 5, 6\}$  e

$E = \{(1, 2), (1, 5), (2, 5), (3, 6), (2, 1), (5, 1), (5, 2), (6, 3)\}$

## Componentes Conectados - Exemplo/Exercício

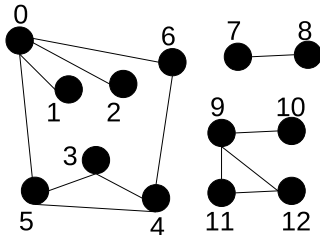
Utilize o algoritmo para encontrar os componentes fortemente conectados nos seguintes grafos:

*Exemplo 1:*

$G(V, E)$  onde  $V = \{1, 2, 3, 4, 5, 6\}$  e

$E = \{(1, 2), (1, 5), (2, 5), (3, 6), (2, 1), (5, 1), (5, 2), (6, 3)\}$

*Exemplo 2:*



## *Componentes Conectados - Exercícios*

1. Altere o algoritmo DFS (Cormem et al.) para encontrar os componentes conectados;
2. Altere o algoritmo DFS (Cormem et al.) para retornar o tamanho (número de vértices) de cada componente conectado.