

# Universidade Federal de Alfenas

## Algoritmos em Grafos

Aula 09 – Árvore Geradora Mínima: Algoritmo de Kruskal

Prof. Humberto César Brandão de Oliveira

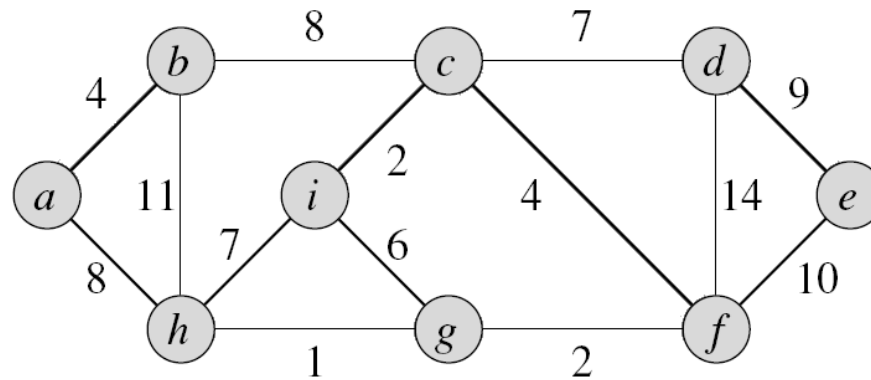
[humberto@bcc.unifal-mg.edu.br](mailto:humberto@bcc.unifal-mg.edu.br)



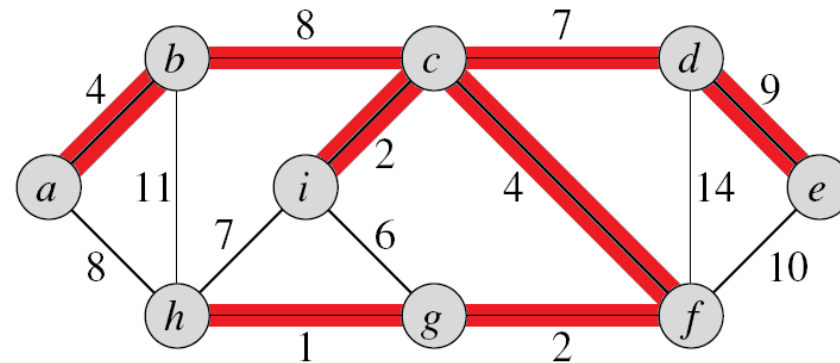
# Relembrando aula passada...

- Exemplo:

$$G = (V, A)$$



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



# Relembrando aula passada...

$AGM\_GENERICA( G(V, A), w )$

$X \leftarrow \{ \}$

enquanto  $|X| \leq |V| - 1$  faça

encontrar uma aresta  $(u, v)$  segura para  $X$

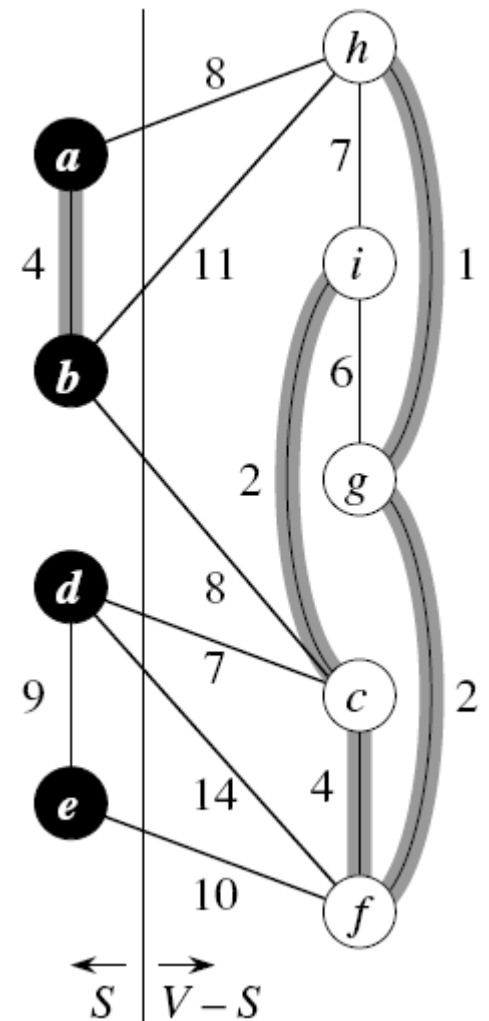
$X \leftarrow X \cup \{(u, v)\}$

fim enquanto

retorna  $X$

fim.

1. Corte
2. Aresta segura
3. Aresta leve



# Árvore Geradora Mínima

- Dois algoritmos clássicos para a AGM:
  - Kruskal;
  - Prim;
- Primeiro algoritmo:
  - Boruvka.

# Árvore Geradora Mínima

- **Prim:**
  - Gera uma árvore única;
  - Ao longo do algoritmo, o conjunto  $X$  sempre é uma árvore.
- **Kruskal:**
  - Gera uma floresta, antes de gerar a AGM;
  - Existe garantia de ser apenas uma árvore apenas depois da última iteração.

# Kruskal

- Na aula de hoje vamos estudar o algoritmo de Kruskal:
  - Criado por Joseph Bernard Kruskal, Jr.
  - Nascido em 1928.
  - Terminou seu PhD na Universidade de Princeton em 1956



# Árvore Geradora Mínima

- Arestas seguras:
  - Prim:
    - *A aresta segura é sempre a aresta de peso mínimo que conecta a árvore a um vértice não presente no conjunto  $X$ .*
  - Kruskal:
    - *A aresta segura é sempre uma aresta de peso mínimo no grafo que conecta dois componentes distintos (duas árvores distintas na floresta).*

# Kruskal

- Ponto Chave:
  - Ele encontra uma **aresta segura** para adicionar à floresta encontrando, **de todas as arestas que conectam duas árvores quaisquer, uma aresta de peso mínimo**;
    - Se você reparar, **o corte acontece neste ponto...** Mas para isso, é efetuada uma adaptação no grafo original
- Kruskal é considerado um **algoritmo guloso**, porque em cada passo ele adiciona à floresta uma **aresta de peso mínimo** (daquelas que ainda podem ser adicionadas).
  - Ou seja, faz uma avaliação dentre todas as possibilidades que possui;



# Kruskal

*AGM \_Kruskal( $G(V, A), w$ )*

*$X \leftarrow \{ \}$*

*para cada vértice  $v \in V$  faça*

*criarConjunto( $v$ )*

*fim para*

*$A' \leftarrow$  ordenar as arestas de  $A$  por peso crescente*

*para cada aresta  $(u, v) \in A'$  faça*

*se conjuntoDe( $u$ )  $\neq$  conjuntoDe( $v$ ) então*

*$X \leftarrow X \cup \{(u, v)\}$*

*aplicarUnião( $u, v$ )*

*fim se*

*fim para*

*retorne  $X$*

*fim.*

# Kruskal

*AGM\_Kruskal( $G(V, A), w$ )*

*$X \leftarrow \{ \}$*

*para cada vértice  $v \in V$  faça*

*criarConjunto( $v$ )*

*fim para*

*$A' \leftarrow$  ordenar as arestas de  $A$  por peso crescente*

*para cada aresta  $(u, v) \in A'$  faça*

*se conjuntoDe( $u$ )  $\neq$  conjuntoDe( $v$ ) então*

*$X \leftarrow X \cup \{(u, v)\}$*

*aplicarUnião( $u, v$ )*

*fim se*

*fim para*

*retorne  $X$*

*fim.*

A princípio, o conjunto que guarda as arestas da AGM é vazio.

# Kruskal

*AGM\_Kruskal( $G(V, A), w$ )*

*$X \leftarrow \{ \}$*

*para cada vértice  $v \in V$  faça*

*criarConjunto( $v$ )*

*fim para*

*$A' \leftarrow$  ordenar as arestas de  $A$  por peso crescente*

*para cada aresta  $(u, v) \in A'$  faça*

*se conjuntoDe( $u$ )  $\neq$  conjuntoDe( $v$ ) então*

*$X \leftarrow X \cup \{(u, v)\}$*

*aplicarUnião( $u, v$ )*

*fim se*

*fim para*

*retorne  $X$*

*fim.*

|V| árvores são criadas.

# Kruskal

*AGM \_Kruskal( $G(V, A), w$ )*

*$X \leftarrow \{ \}$*

*para cada vértice  $v \in V$  faça*

*criarConjunto( $v$ )*

*fim para*

*$A' \leftarrow$  ordenar as arestas de  $A$  por peso crescente*

*para cada aresta  $(u, v) \in A'$  faça*

*se conjuntoDe( $u$ )  $\neq$  conjuntoDe( $v$ ) então*

*$X \leftarrow X \cup \{(u, v)\}$*

*aplicarUnião( $u, v$ )*

*fim se*

*fim para*

*retorne  $X$*

*fim.*

O conjunto de arestas é ordenado em função dos pesos. Condição necessária para a criação da AGM através de Kruskal

# Kruskal

*AGM\_Kruskal( $G(V, A), w$ )*

*$X \leftarrow \{ \}$*

*para cada vértice  $v \in V$  faça*

*criarConjunto( $v$ )*

*fim para*

*$A' \leftarrow$  ordenar as arestas de  $A$  por peso crescente*

*para cada aresta  $(u, v) \in A'$  faça*

*se conjuntoDe( $u$ )  $\neq$  conjuntoDe( $v$ ) então*

*$X \leftarrow X \cup \{(u, v)\}$*

*aplicarUnião( $u, v$ )*

*fim se*

*fim para*

*retorne  $X$*

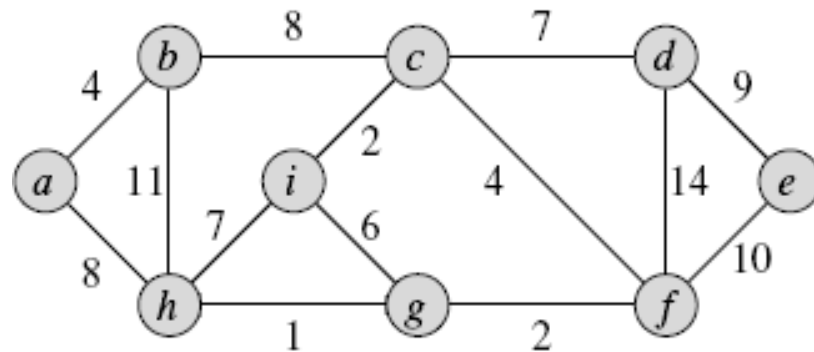
*fim.*

Para cada aresta  
do vetor  
ordenado

Se  $u$  e  $v$  são de  
árvores distintas,  
a aresta  $(u, v)$  é  
adicionada ao  
conjunto  $X$  e é  
aplicada uma  
união das  
árvores de  $u$  e  $v$ .

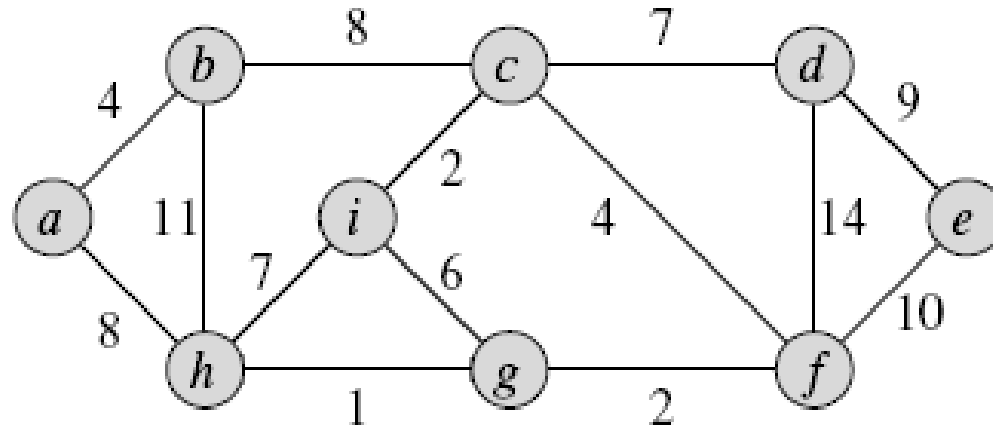
# AGM utilizando Kruskal

- Considerando o grafo a seguir... Vamos criar passo-a-passo a AGM utilizando Kruskal...



# AGM utilizando Kruskal

- 1º passo: criar um(a) conjunto/árvore para cada vértice.

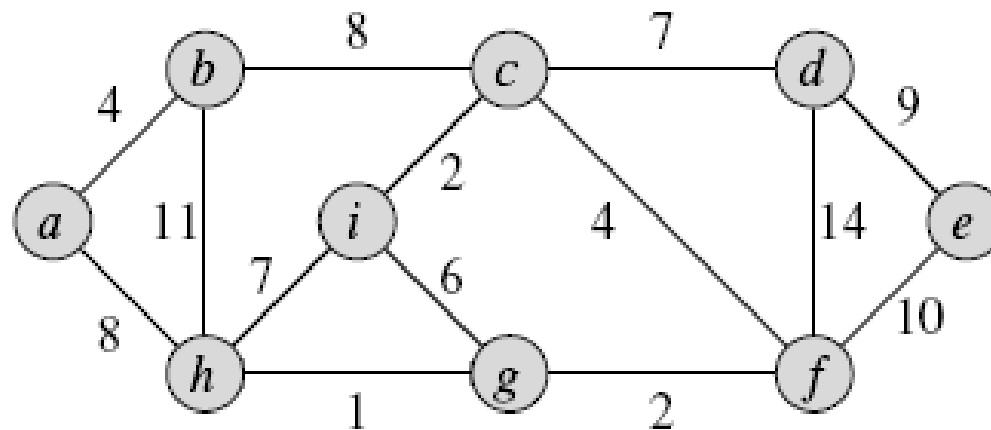


$\{\{a\}, \{b\}, \{c\}, \{d\}, \{e\}, \{f\}, \{g\}, \{h\}, \{i\}\}$

# AGM utilizando Kruskal

- 2º passo: ordenar as arestas do conjunto A.

$\{\{a\}, \{b\}, \{c\}, \{d\}, \{e\}, \{f\}, \{g\}, \{h\}, \{i\}\}$



$A'$   $(g, h); (c, i); (f, g); (a, b); (c, f); (g, i); (c, d); (h, i);$   
 $(a, h); (b, c); (d, e); (e, f); (b, h); (d, f)$

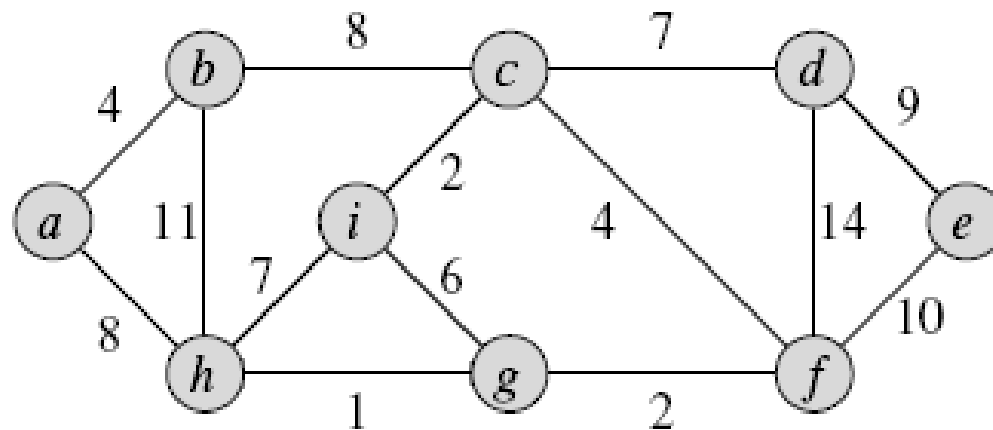


# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$\{\{a\}, \{b\}, \{c\}, \{d\}, \{e\}, \{f\}, \{g\}, \{h\}, \{i\}\}$

g e h pertencem a mesma árvore na floresta?



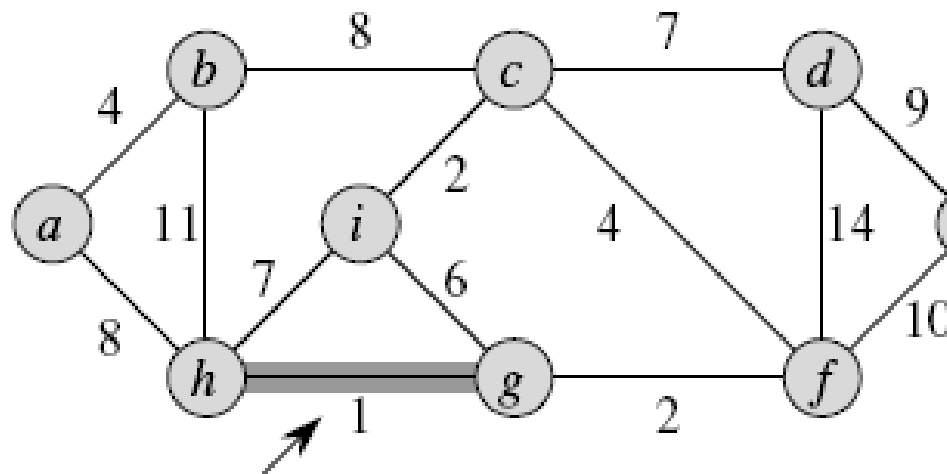
A'  $(g, h); (c, i); (f, g); (a, b); (c, f); (g, i); (c, d); (h, i);$   
 $(a, h); (b, c); (d, e); (e, f); (b, h); (d, f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$\{\{a\}, \{b\}, \{c\}, \{d\}, \{e\}, \{f\}, \{g, h\}, \{i\}\}$

g e h pertencem a mesma árvore na floresta?



Não... Então...  
 União das árvores de g e h e adição da aresta na AGM

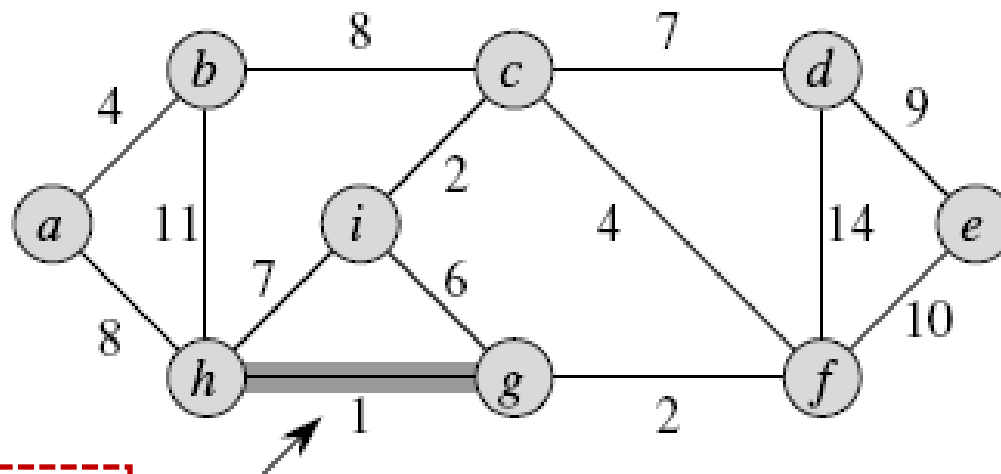
A'  $(g, h); (c, i); (f, g); (a, b); (c, f); (g, i); (c, d); (h, i);$   
 $(a, h); (b, c); (d, e); (e, f); (b, h); (d, f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$\{\{a\}, \{b\}, \{c\}, \{d\}, \{e\}, \{f\}, \{g, h\}, \{i\}\}$

*c e i pertencem a mesma árvore na floresta?*



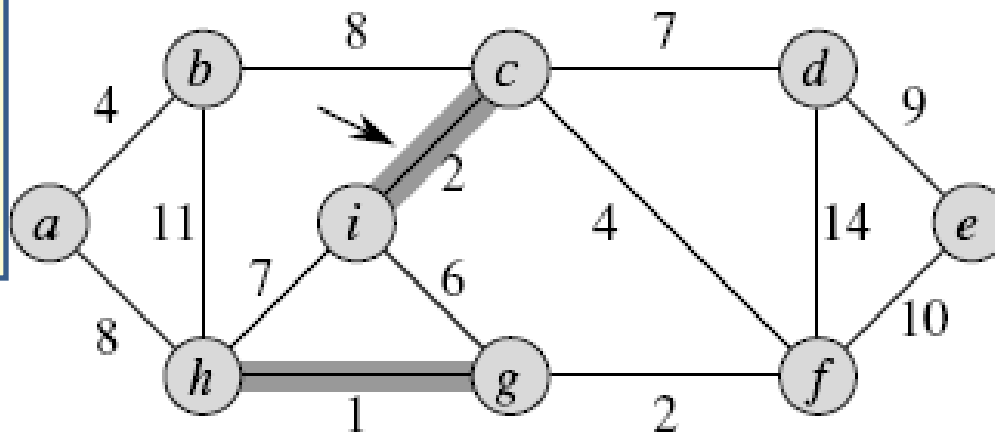
$A' \quad (g, h); (c, i); (f, g); (a, b); (c, f); (g, i); (c, d); (h, i);$   
 $(a, h); (b, c); (d, e); (e, f); (b, h); (d, f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$\{\{a\}, \{b\}, \{c, i\}, \{d\}, \{e\}, \{f\}, \{g, h\}\}$

Não... Então...  
União das árvores  
de  $c$  e  $i$  e adição da  
aresta na AGM



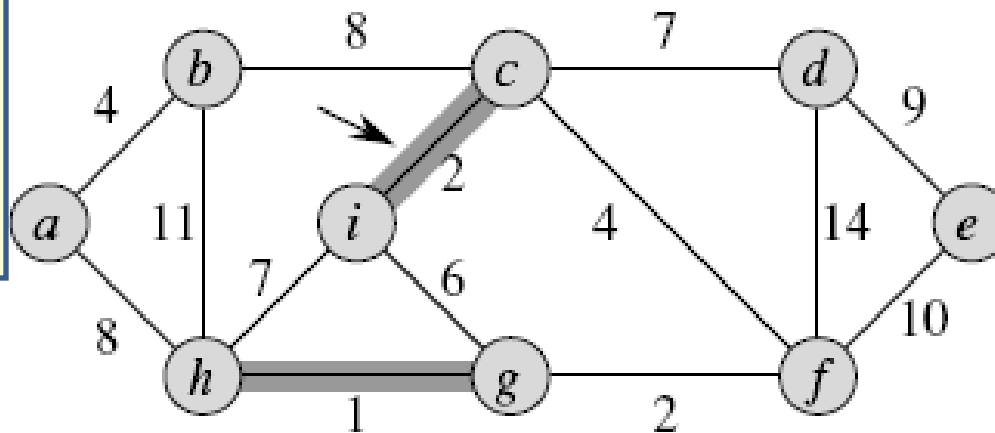
$A' \quad (g, h); (c, i); (f, g); (a, b); (c, f); (g, i); (c, d); (h, i);$   
 $(a, h); (b, c); (d, e); (e, f); (b, h); (d, f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$\{\{a\}, \{b\}, \{c, i\}, \{d\}, \{e\}, \{f\}, \{g, h\}\}$

*f e g pertencem a mesma árvore na floresta?*



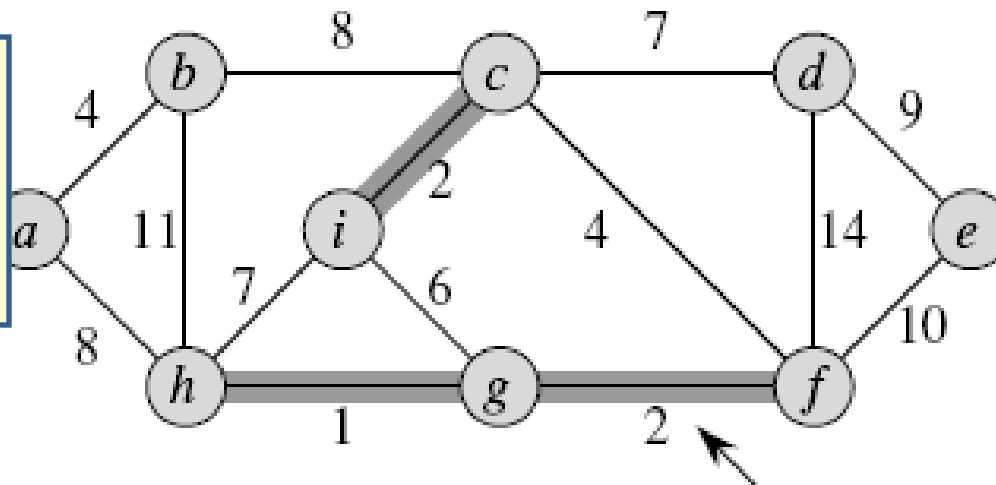
$A' \quad (g, h); (c, i); (f, g); (a, b); (c, f); (g, i); (c, d); (h, i);$   
 $(a, h); (b, c); (d, e); (e, f); (b, h); (d, f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$\{\{a\}, \{b\}, \{c, i\}, \{d\}, \{e\}, \{f, g, h\}\}$

Não... Então...  
 União das árvores  
 de  $f$  e  $g$  e adição da  
 aresta na AGM



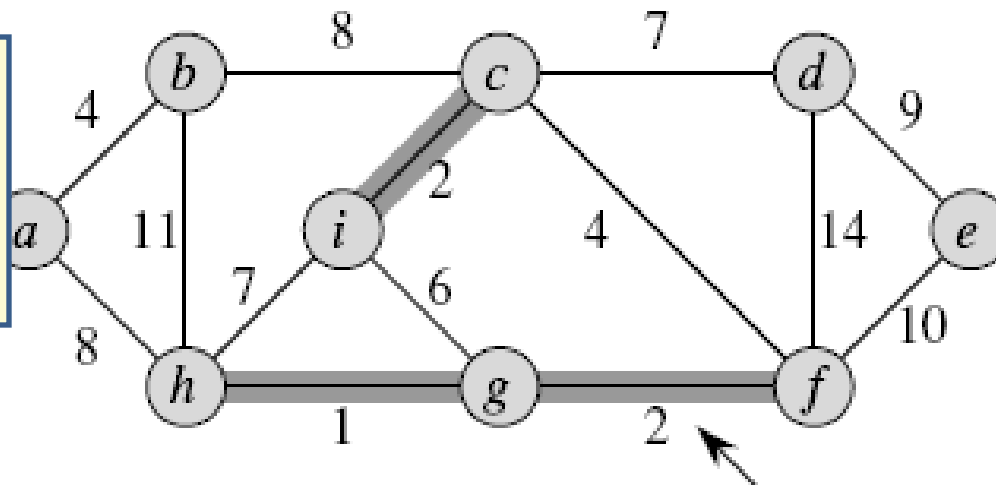
$A' \quad (g, h); (c, i); (f, g); (a, b); (c, f); (g, i); (c, d); (h, i);$   
 $(a, h); (b, c); (d, e); (e, f); (b, h); (d, f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$$\{\{a\}, \{b\}, \{c, i\}, \{d\}, \{e\}, \{f, g, h\}\}$$

*a e b pertencem a mesma árvore na floresta?*

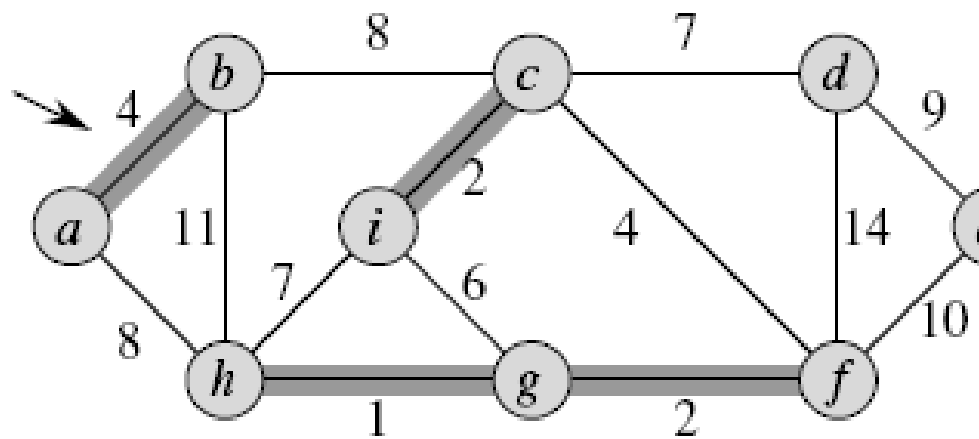


$A'$   $(g, h); (c, i); (f, g); (a, b); (c, f); (g, i); (c, d); (h, i);$   
 $(a, h); (b, c); (d, e); (e, f); (b, h); (d, f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$\{\{a,b\}, \{c,i\}, \{d\}, \{e\}, \{f,g,h\}\}$



Não... Então...  
 União das árvores  
 de  $a$  e  $b$  e adição da  
 aresta na AGM

$A'$   $(g,h); (c,i); (f,g); (a,b); (c,f); (g,i); (c,d); (h,i);$   
 $(a,h); (b,c); (d,e); (e,f); (b,h); (d,f)$

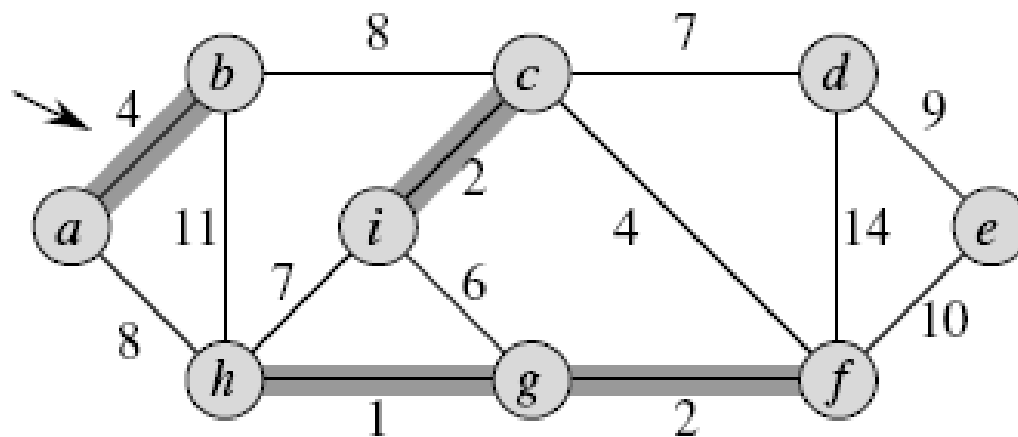


# AGM utilizando Kruskal

*c e f pertencem a mesma árvore na floresta?*

- 3º passo: para cada aresta ordenada, faç

$\{\{a,b\}, \{c,i\}, \{d\}, \{e\}, \{f,g,h\}\}$

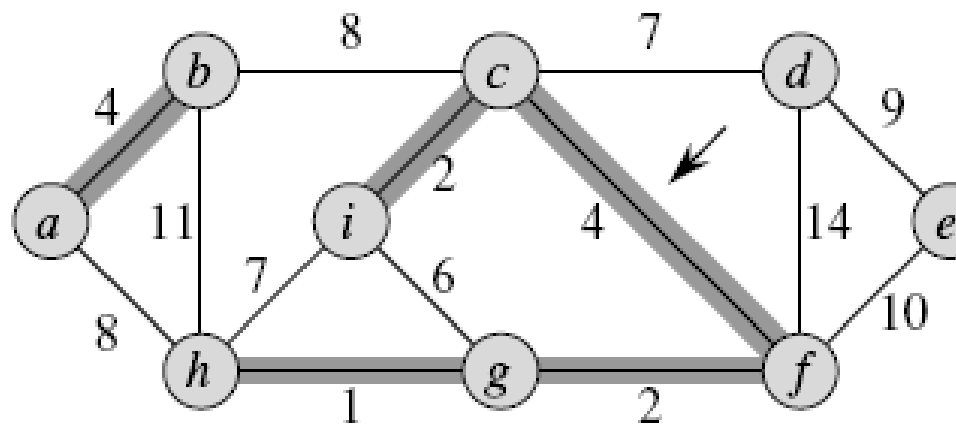


$A' \quad (g,h); (c,i); (f,g); (a,b); (c,f); (g,i); (c,d); (h,i);$   
 $(a,h); (b,c); (d,e); (e,f); (b,h); (d,f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$\{\{a,b\}, \{d\}, \{e\}, \{c, f, g, h, i\}\}$



Não... Então...  
 União das árvores  
 de  $c$  e  $f$  e adição da  
 aresta na AGM

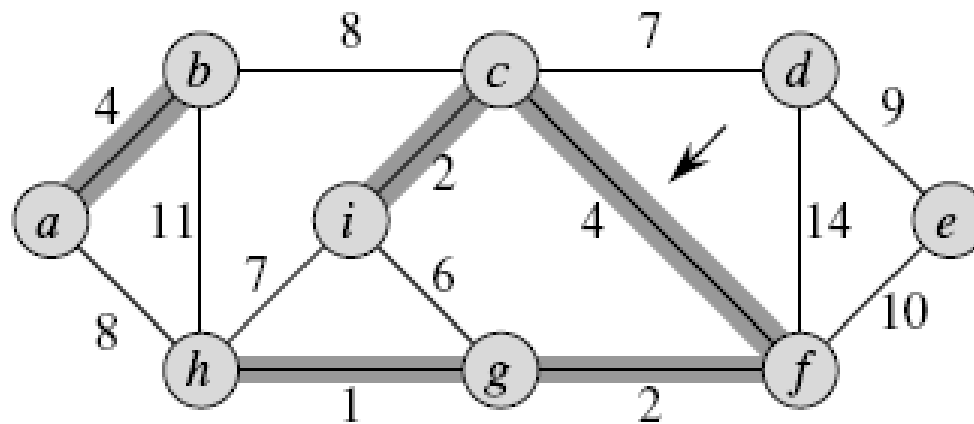
$A'$   $(g,h); (c,i); (f,g); (a,b); (c,f); (g,i); (c,d); (h,i);$   
 $(a,h); (b,c); (d,e); (e,f); (b,h); (d,f)$

# AGM utilizando Kruskal

*g e i pertencem a mesma árvore na floresta?*

- 3º passo: para cada aresta ordenada, faç

$\{\{a,b\}, \{d\}, \{e\}, \{c, f, g, h, i\}\}$



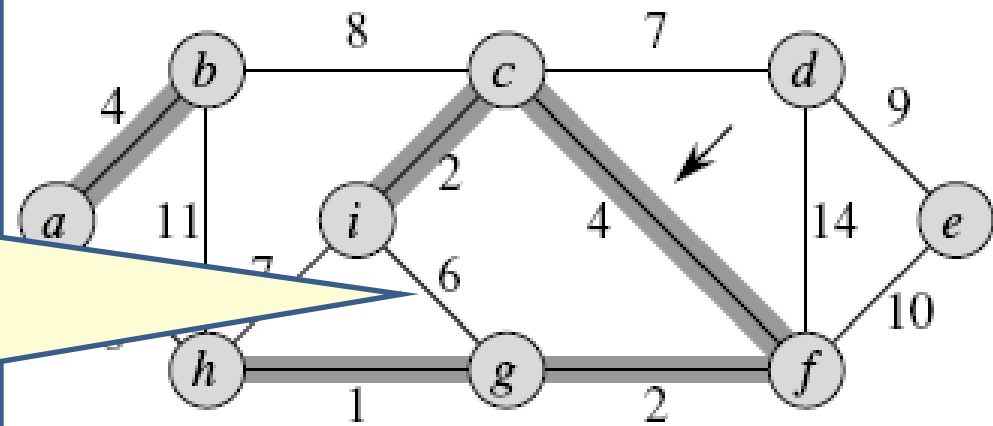
$A' \quad (g,h); (c,i); (f,g); (a,b); (c,f); (g,i); (c,d); (h,i);$   
 $(a,h); (b,c); (d,e); (e,f); (b,h); (d,f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$\{\{a,b\}, \{d\}, \{e\}, \{c, f, g, h, i\}\}$

*(g,i) fecha um ciclo. Isso é identificado porque g e i pertencem a mesma árvore na estrutura auxiliar 'floresta'.*

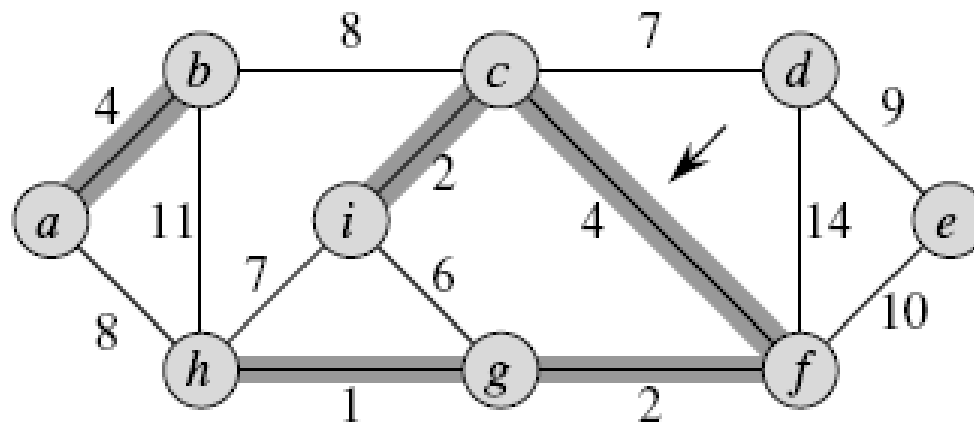


A'  $(g, h); (c, i); (f, g); (a, b); (c, f); (\cancel{g, i}); (c, d); (h, i);$   
 $(a, h); (b, c); (d, e); (e, f); (b, h); (d, f)$

# AGM utilizando Kruskal

*c e d pertencem a mesma árvore na floresta?*

- 3º passo: para cada aresta ordenada, faç

$$\{\{a,b\}, \{d\}, \{e\}, \{c, f, g, h, i\}\}$$


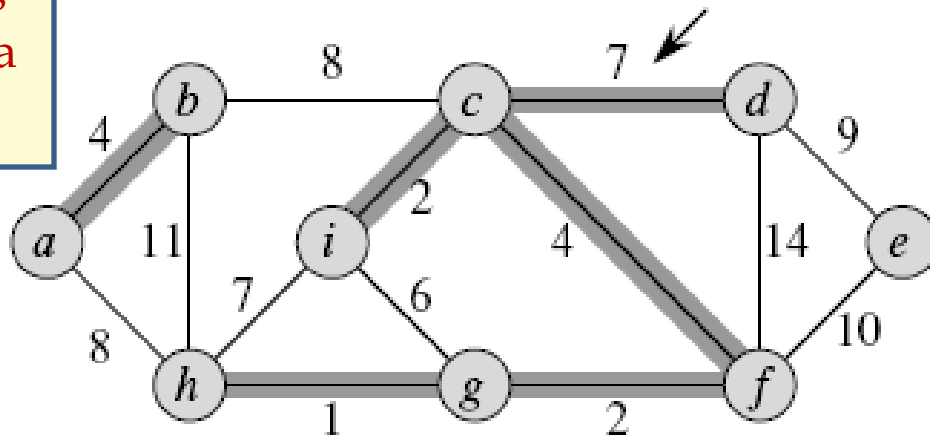
$A'$   $(g, h); (c, i); (f, g); (a, b); (c, f); (g, i); (c, d); (h, i);$   
 $(a, h); (b, c); (d, e); (e, f); (b, h); (d, f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$\{\{a,b\}, \{e\}, \{c,d,f,g,h,i\}\}$

Não... Então...  
 União das árvores  
 de  $c$  e  $d$  e adição da  
 aresta na AGM



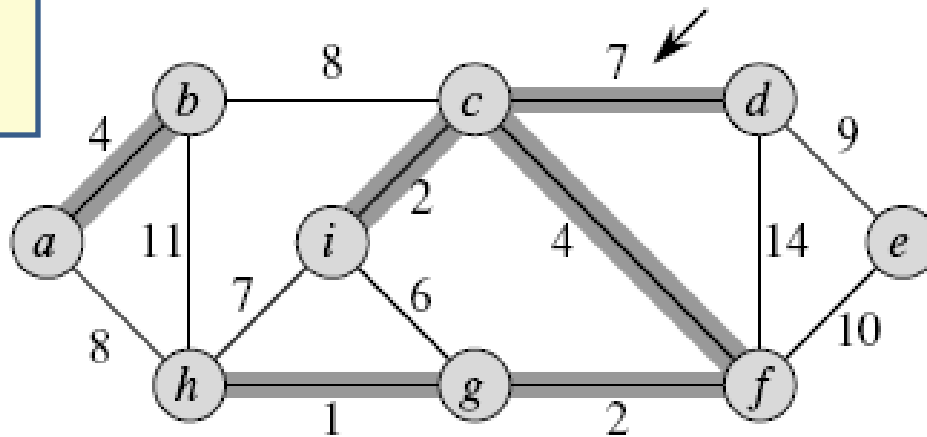
$A'$   $(g,h); (c,i); (f,g); (a,b); (c,f); (g,i); (c,d); (h,i);$   
 $(a,h); (b,c); (d,e); (e,f); (b,h); (d,f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

*h e i pertencem a  
mesma árvore na  
floresta?*

$\{\{a,b\}, \{e\}, \{c,d,f,g,h,i\}\}$



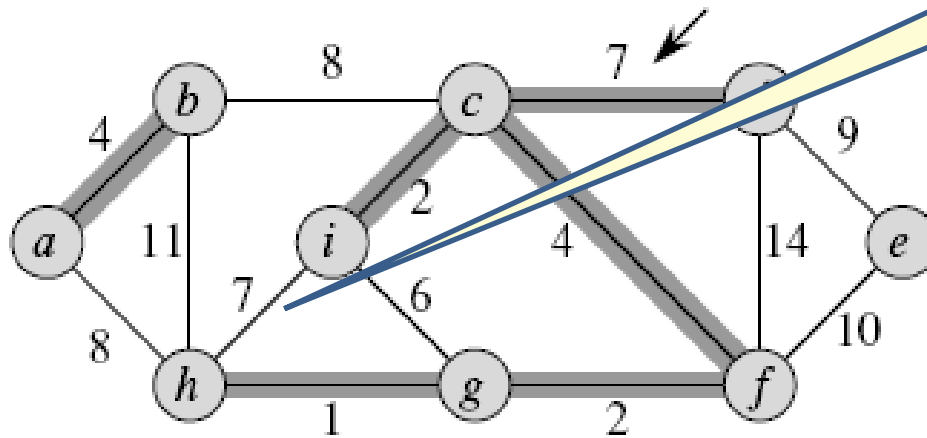
$A'$   $(g,h); (c,i); (f,g); (a,b); (c,f); (g,i); (c,d); (h,i);$   
 $(a,h); (b,c); (d,e); (e,f); (b,h); (d,f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$\{\{a,b\}, \{e\}, \{c,d,f,g,h,i\}\}$

*Fecha ciclo...*



$A'$   $(g,h); (c,i); (f,g); (a,b); (c,f); (\cancel{g,i}); (c,d); (\cancel{h,i});$   
 $(a,h); (b,c); (d,e); (e,f); (b,h); (d,f)$

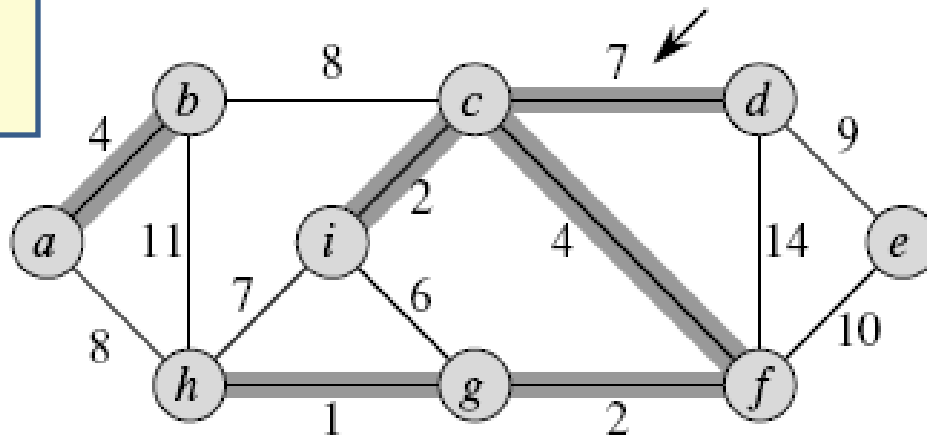


# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

*a e h pertencem a mesma árvore na floresta?*

$\{\{a,b\}, \{e\}, \{c,d,f,g,h,i\}\}$



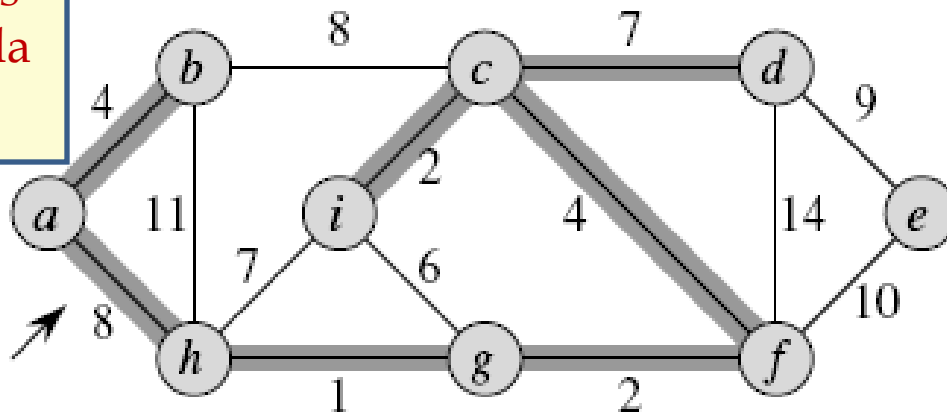
$A' \quad (g,h); (c,i); (f,g); (a,b); (c,f); (\cancel{g,i}); (\cancel{c,d}); (\cancel{h,i});$   
 $(a,h); (b,c); (d,e); (e,f); (b,h); (d,f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$\{\{e\}, \{a, b, c, d, f, g, h, i\}\}$

Não... Então...  
 União das árvores  
 de *a* e *h* e adição da  
 aresta na AGM



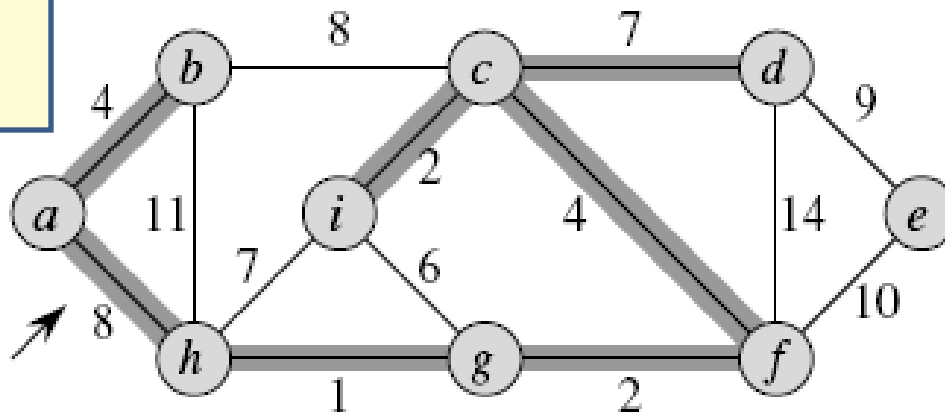
$A'$   $(g, h); (c, i); (f, g); (a, b); (c, f); (\cancel{g, i}); (\cancel{c, d}); (\cancel{h, i});$   
 $(a, h); (b, c); (d, e); (e, f); (b, h); (d, f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

*b e c pertencem a mesma árvore na floresta?*

$\{\{e\}, \{a, b, c, d, f, g, h, i\}\}$

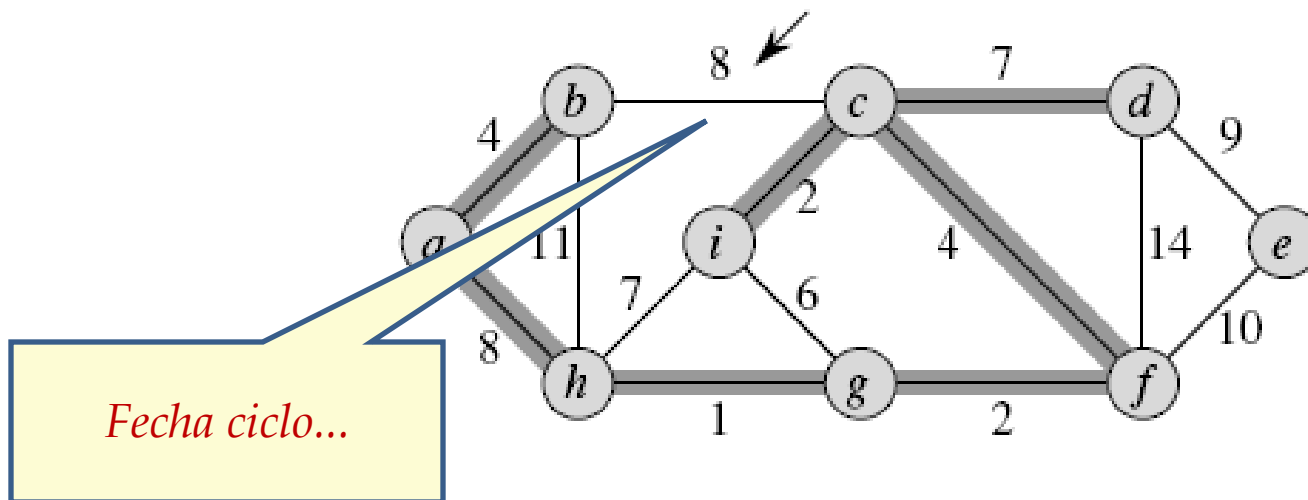


$A'$   $(g, h); (c, i); (f, g); (a, b); (c, f); (\cancel{g, i}); (\cancel{c, d}); (\cancel{h, i});$   
 $(a, h); (b, c); (d, e); (e, f); (b, h); (d, f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$\{\{e\}, \{a, b, c, d, f, g, h, i\}\}$

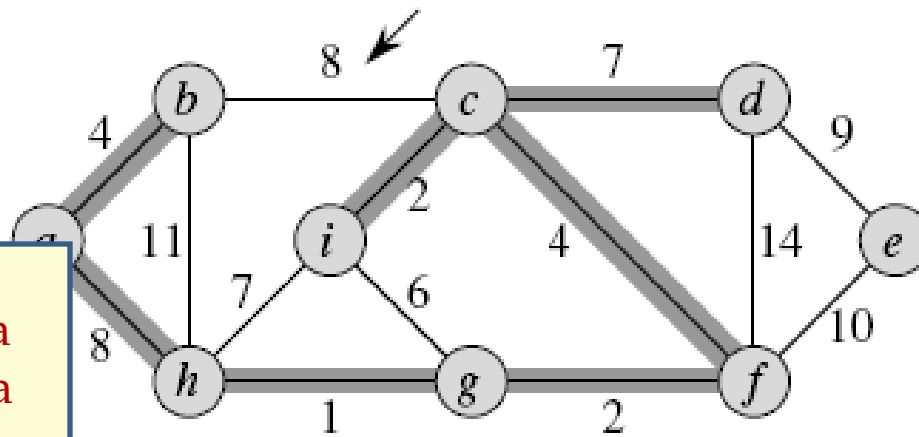


$A'$   $(g, h); (c, i); (f, g); (a, b); (c, f); (\cancel{g, i}); (\cancel{c, d}); (\cancel{h, i});$   
 $(a, h); (\cancel{b, c}); (d, e); (e, f); (b, h); (d, f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$$\{\{e\}, \{a, b, c, d, f, g, h, i\}\}$$



*d e e pertencem a mesma árvore na floresta?*

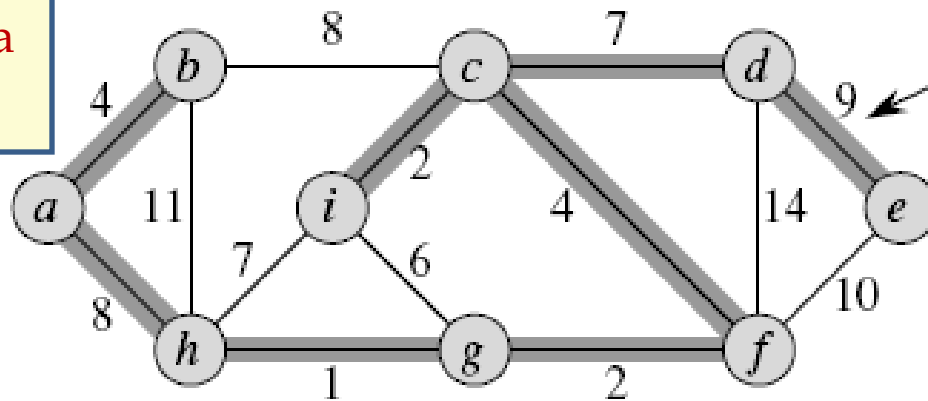
$A'$   $(g, h); (c, i); (f, g); (a, b); (c, f); (\cancel{g, i}); (\cancel{c, d}); (\cancel{h, i});$   
 $(a, h); (\cancel{b, c}); (d, e); (e, f); (b, h); (d, f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$\{a, b, c, d, e, f, g, h, i\}$

Não... Então...  
 União das árvores  
 de  $d$  e  $e$  e adição da  
 aresta na AGM



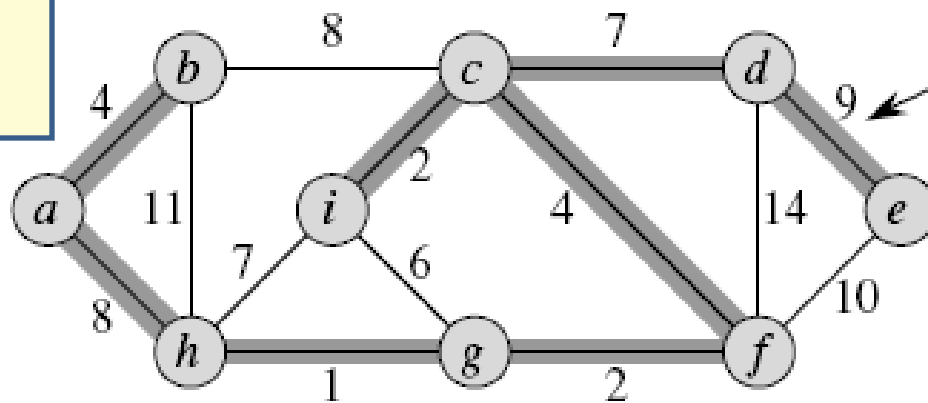
$A'$   $(g, h); (c, i); (f, g); (a, b); (c, f); (\cancel{g, i}); (c, d); (\cancel{h, i});$   
 $(a, h); (\cancel{b, c}); (d, e); (e, f); (b, h); (d, f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

*e e f pertencem a mesma árvore na floresta?*

$\{a, b, c, d, e, f, g, h, i\}$



$A'$   $(g, h); (c, i); (f, g); (a, b); (c, f); (\cancel{g, i}); (c, d); (\cancel{h, i});$   
 $(a, h); (\cancel{b, c}); (d, e); (e, f); (b, h); (d, f)$



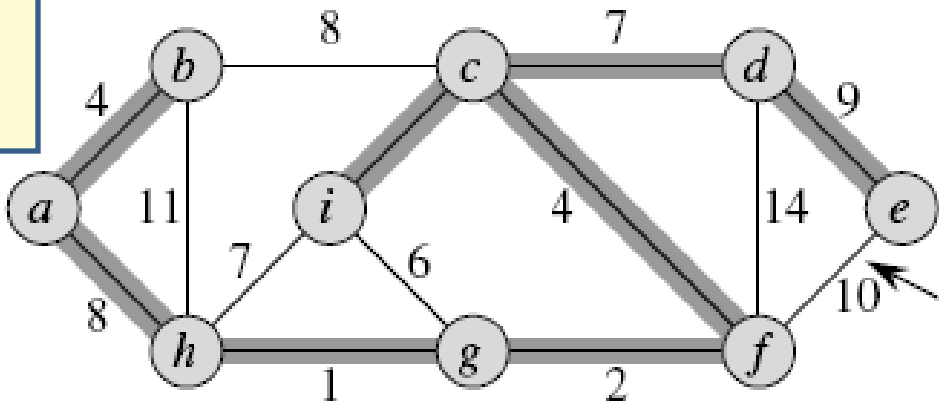


# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

*b e h pertencem a mesma árvore na floresta?*

$\{a, b, c, d, e, f, g, h, i\}$



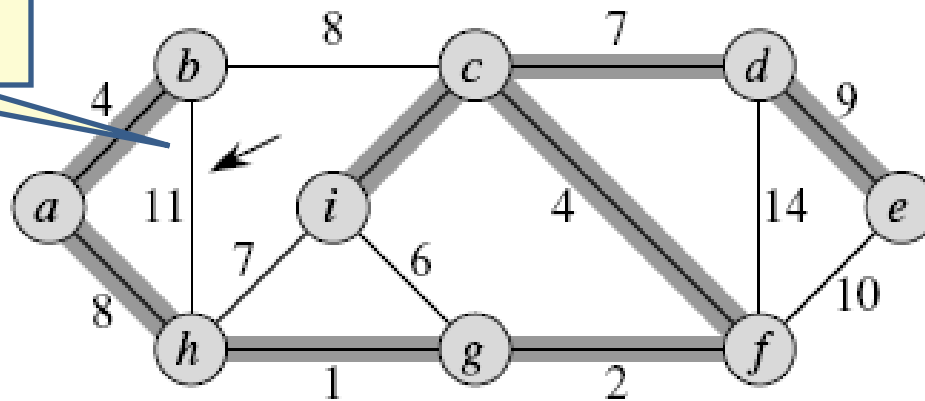
$A'$   $(g, h); (c, i); (f, g); (a, b); (c, f); (\cancel{g, i}); (c, d); (\cancel{h, i});$   
 $(a, h); (\cancel{b, c}); (d, e); (\cancel{e, f}); (b, h); (d, f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$\{a, b, c, d, e, f, g, h, i\}$

Fecha ciclo...

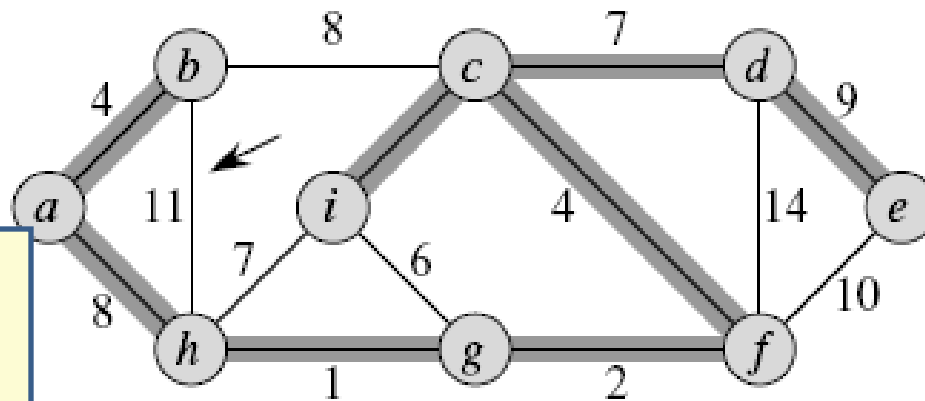


$A'$   $(g, h); (c, i); (f, g); (a, b); (c, f); (\cancel{g, i}); (c, d); (\cancel{h, i});$   
 $(a, h); (\cancel{b, c}); (d, e); (\cancel{e, f}); (\cancel{b, h}); (d, f)$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$\{\{a, b, c, d, e, f, g, h, i\}\}$



*d e f pertencem a mesma árvore na floresta?*

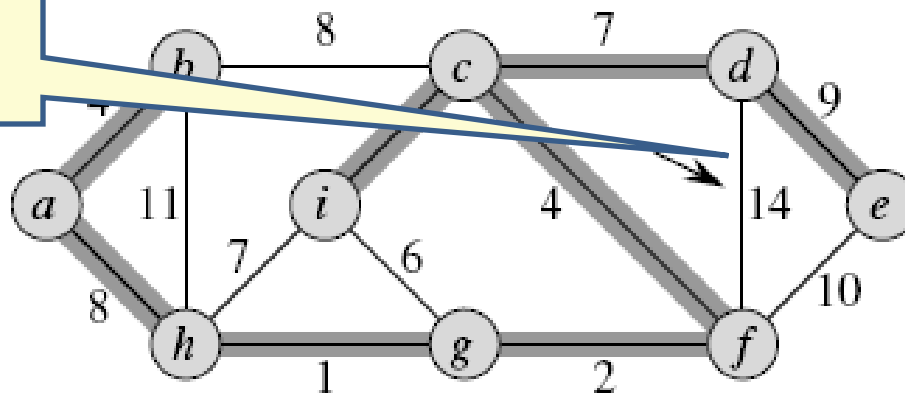
$A'$   $(g, h); (c, i); (f, g); (a, b); (c, f); (\cancel{g, i}); (c, d); (\cancel{h, i});$   
 $(a, h); (\cancel{b, c}); (d, e); (\cancel{e, f}); (\cancel{b, h}); (\cancel{d, f})$

# AGM utilizando Kruskal

- 3º passo: para cada aresta ordenada, faça...

$\{a, b, c, d, e, f, g, h, i\}$

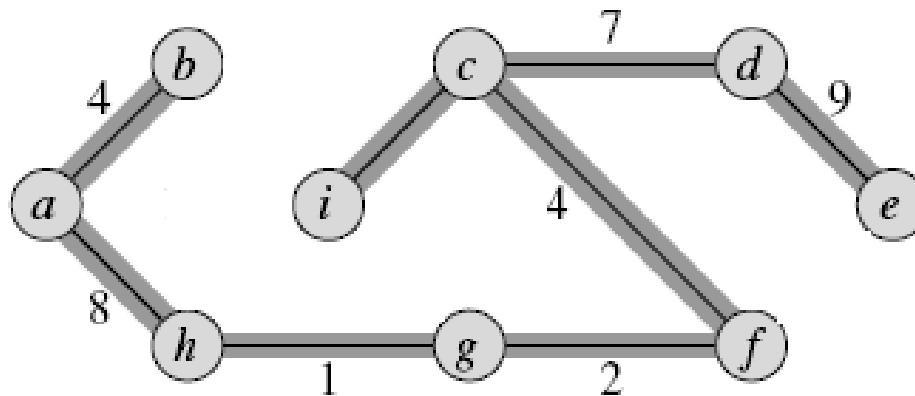
*Fecha ciclo...*



$A'$   $(g, h); (c, i); (f, g); (a, b); (c, f);$  ~~$(g, i); (c, d); (h, i);$~~   
 ~~$(a, h); (b, c); (d, e); (e, f); (b, h); (d, f)$~~

# AGM utilizando Kruskal

- O conjunto X (arestas da AGM) foi composto ao longo da execução do Kruskal, onde apenas as arestas não marcadas de A' foram adicionadas à árvore.



$A'$   $(g, h); (c, i); (f, g); (a, b); (c, f); (\cancel{g, i}); (c, d); (\cancel{h, i});$   
 $(a, h); (\cancel{b, c}); (d, e); (\cancel{e, f}); (\cancel{b, h}); (\cancel{d, f})$

# Animação na Web do algoritmo de Kruskal

- Pode ser feito passo a passo. Bom para entendimento geral do algoritmo:
  - <http://students.ceid.upatras.gr/~papagel/project/kruskal.htm>

# Exercício

- Qual é a complexidade do algoritmo de Kruskal?
- Ela depende de quais estruturas?
- Proponha uma estrutura eficiente para armazenar e efetuar as operações na floresta e nas árvores do algoritmo de Kruskal.

# Bibliografia

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

