

Universidade Federal do Piauí – UFPI

Campus Senador Helvídio Nunes de Barros – CSHNB

Curso de Sistemas de Informação Bloco: IV

Disciplina: Estruturas de Dados II

Professora: Juliana Oliveira de Carvalho

Acadêmico: **Elievelto Edimar da Silva** Matrícula: **20179018249**

### ATIVIDADE DE FIXAÇÃO 9

- 1) Identifique todas as árvores geradoras dos grafos das figuras a seguir:

Para identificar uma árvore geradora de um grafo é preciso retirar suas arestas até não conter nenhum ciclo, e mantendo os mesmos nós.

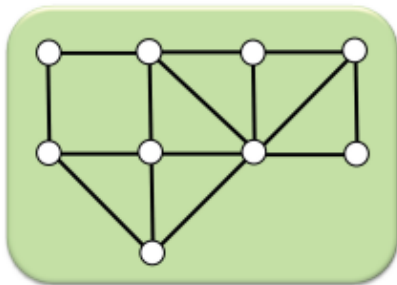
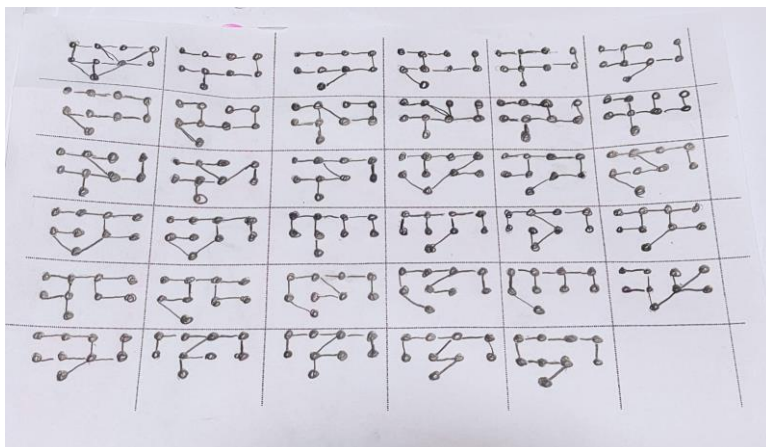


Figura 1



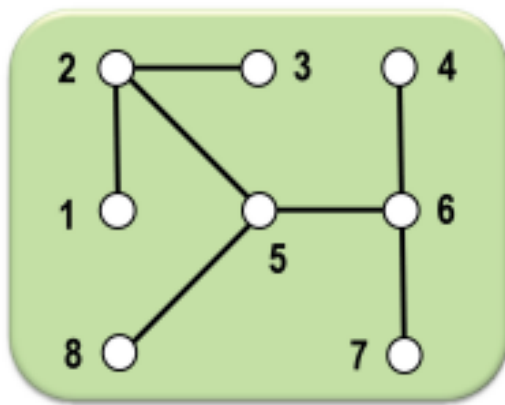


Figura 2

A árvore geradora do grafo acima é o próprio grafo, pois o grafo não possui ciclos, diferente do que fizemos na primeira figura que possuía ciclos.

2) Dados os grafos e as árvores abaixo, determine suas coárvores geradoras

Figura1

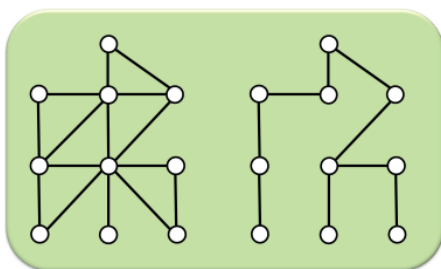
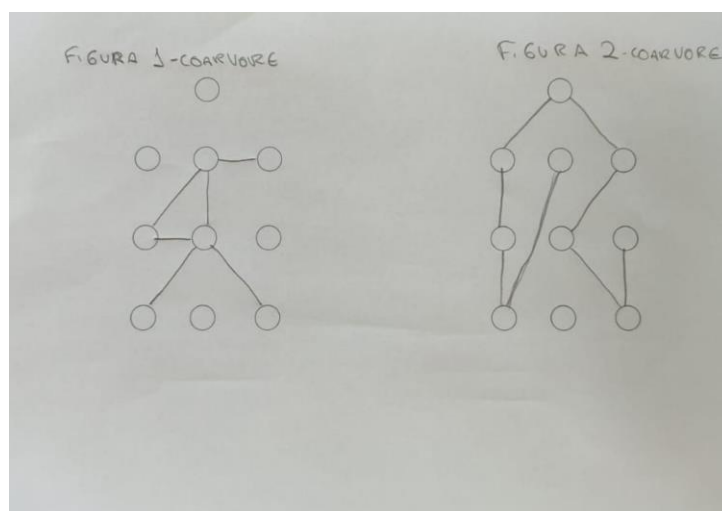
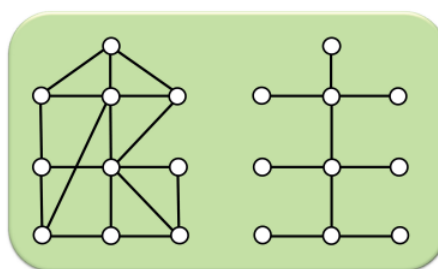


figura2



- 3) Execute os algoritmos Prim, Kruskal e Boruvka par determinação da árvore geradora mínima no grafo da figura a seguir:

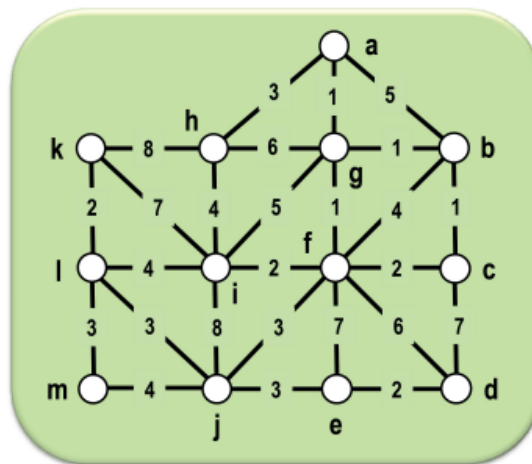


Figura1

**Algoritmo Prim:** Serve para montar uma árvore geradora mínima, o algoritmo de **prim** escolhe um vértice inicial e conecta a ele o vértice com menor pesos de aresta, passando por todos os vértices até que todos estejam interligados

$$H \rightarrow 0$$

$$H - A \rightarrow 0 + 3$$

$$H - A - G \rightarrow 3 + 1$$

$$H - A - G - B \rightarrow 4 + 1$$

$$H - A - G - B - C \rightarrow 5 + 1$$

$$H - A - G - B - C - F \rightarrow 6 + 2$$

$$H - A - G - B - C - F - I \rightarrow 8 + 2$$

$$H - A - G - B - C - F - I - J \rightarrow 10 + 3$$

$$H - A - G - B - C - F - I - J - E \rightarrow 13 + 3$$

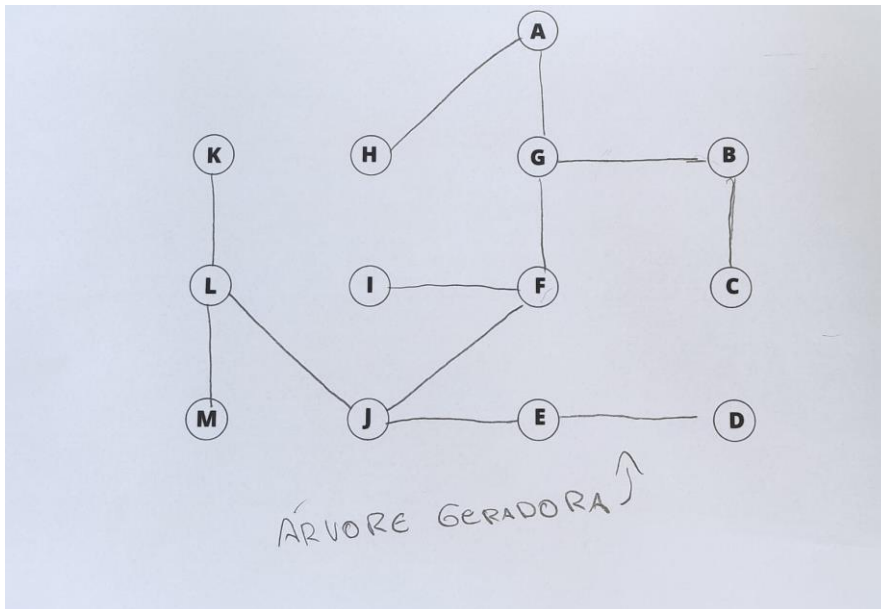
$$H - A - G - B - C - F - I - J - E - D \rightarrow 16 + 2$$

$$H - A - G - B - C - F - I - J - E - D - L \rightarrow 18 + 3$$

$$H - A - G - B - C - F - I - J - E - D - L - K \rightarrow 21 + 2$$

$$H - A - G - B - C - F - I - J - E - D - L - K - M \rightarrow \underline{23 + 3}$$

Custo 26



**Algoritmo Kruskal:** Consiste em dividir o grafo em arestas e ordená-las em ordem crescente de custo.

Sendo assim, o conjunto de arestas do grafo da figura 1 é:

**Conjunto A**

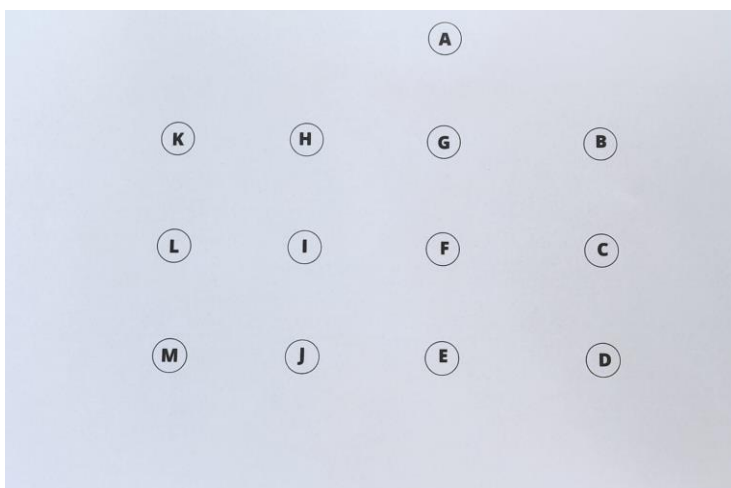
V1: A G G B F I K E A J J L L M L F H A I H F F C K F

V2: G F B C C F L D H F E J M J I B I B G G D E D I H

P: 1 1 1 1 2 2 2 2 3 3 3 3 3 4 4 4 4 5 5 6 6 7 7 7 8

Onde: V1 = Vértice 1, V2 = Vértice 2 e P = Peso das arestas que ligam V1 com V2.

Feito isso devemos criar uma floresta com todos os vértices, sendo cada vértice uma árvore individual, como mostra na figura abaixo:



Floresta contendo todos os vértices

Verificamos também todo o conjunto e perguntamos se cada aresta está dentro da mesma árvore, se sim, não faz nada e continua para a próxima aresta, se não, podemos conectar os nós criando outra árvore. Ex:

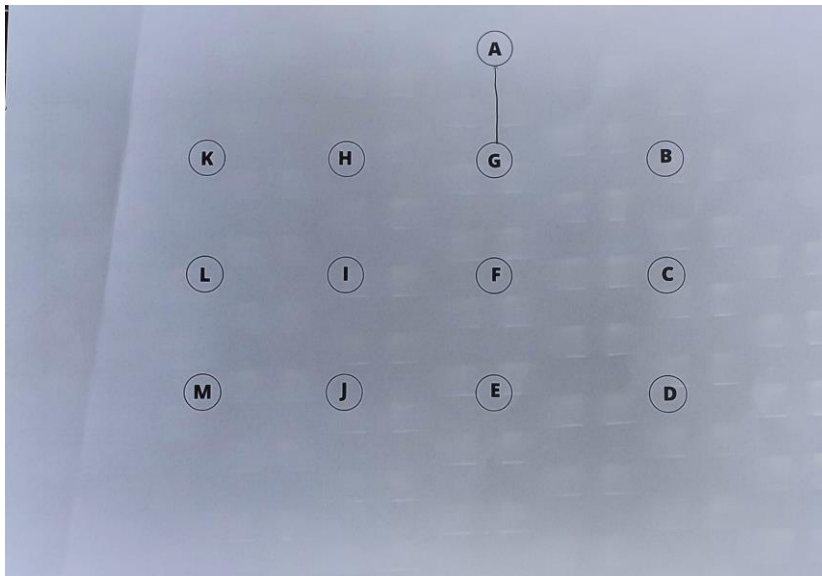
$V1 = A$

$V2 = G \ V1$

Está na mesma árvore que V2?

**Resposta: Não**

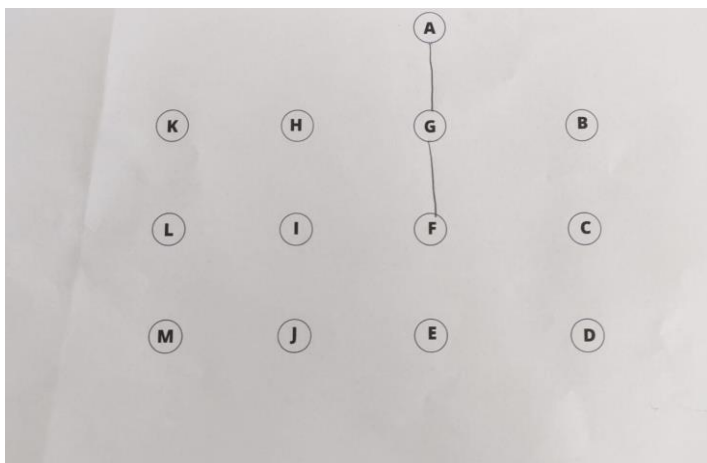
Podemos sim, ligar A com G, como podemos observar na figura abaixo:



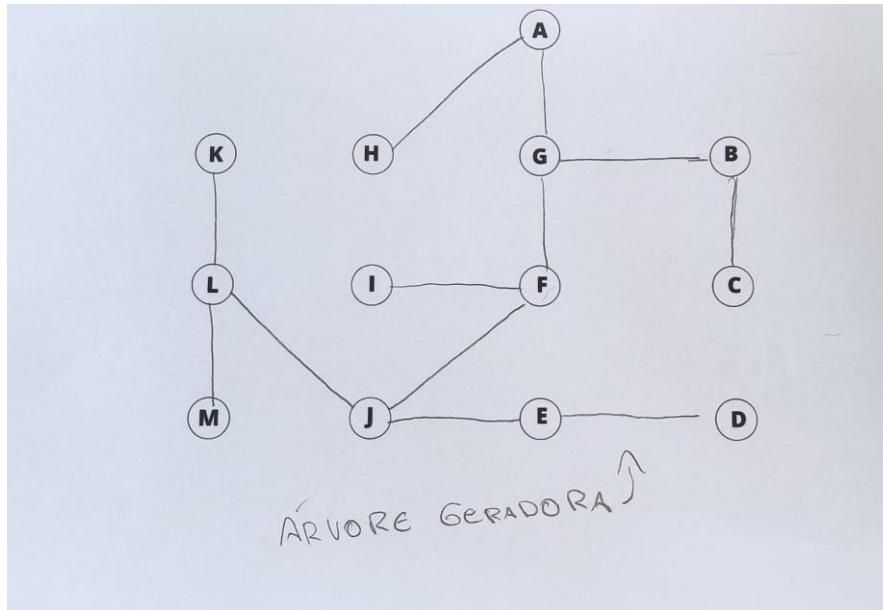
E verificamos os próximos vértices.

G está na mesma árvore que F? Não

Podemos ligá-los.

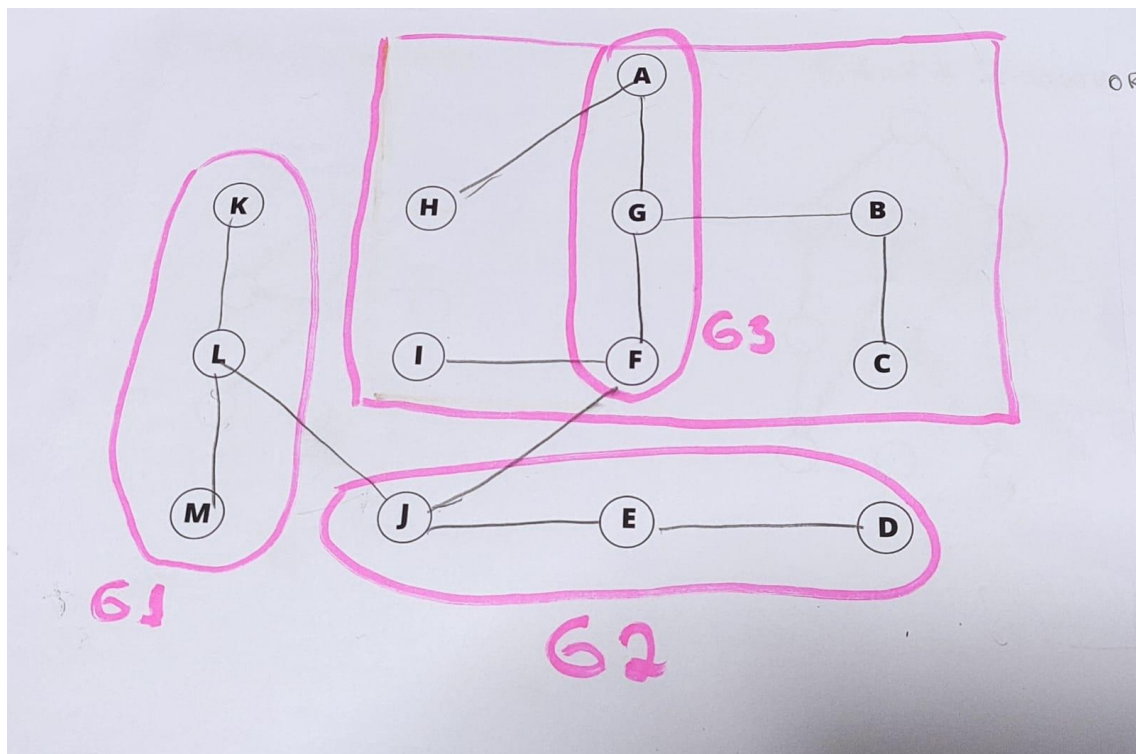


Fazemos isso para todos os vértices do conjunto. Ao final, teremos a árvore geradora mínima para o grafo da figura 5.



Árvore geradora mínima

**Algoritmo Boruvka:** Esse algoritmo analisa cada vértice e se conecta ao seu vizinho com aresta de menor valor formando grupos de sub-grafos, neste nosso exemplo seria criado 3 grupos que seria Grupo 1 (K,L,M), Grupo 2 (J,E,D) e Grupo 3 (H,A,G,F,I,B,C).



Em seguida, cada grupo é ligado seguindo a mesma lógica. (conectando os vértices de menor custo na aresta).

E como resultado final teremos:

