# Capítulo 3
## Árvores

**Problema:** Suponha que numa cidade haja $n$ postos telefônicos. Para que seja sempre possível haver comunicação (não necessariamente direta) entre quaisquer desses postos, qual é o *número mínimo de linhas diretas* que deve existir?

**Pergunta**: Qual é o número mínimo de arestas que um grafo com $n$ vértices deve ter para ser conexo?
**Resposta:**

Vimos no Exercício 19 do Capítulo 1 que

Se um grafo conexo tem $n$ vértices, então tem pelo menos $n-1$ arestas.

Vejamos uma prova desse fato.

**Proposição 3.0.** Se $G$ é um grafo conexo com $n$ vértices, então $A(G)| \geq n - 1$.

**Prova** (na aula).

Ou seja, conforme a Proposição 3.0, temos que $|A(G)| \geq n-1$ é **condição necessária** para que um grafo $G$ com $n$ vértices seja conexo.

**Pergunta 1:** $|A(G)| \geq n - 1$ é **condição suficiente** para garantir que um grafo $G$ com $n$ vértices seja conexo?
**Resposta 1:**

Justificativa:

**Pergunta 2:** Existem grafos conexos com $n$ vértices e $n-1$ arestas, para todo $n \geq 1$?
**Resposta 2:**

Justificativa:

EXERCÍCIO A. Desenhe todos os grafos conexos (não-isomorfos) com $n$ vértices e $n-1$ arestas para $n = 1, 2, \ldots, 6$.

Def. Dizemos que um grafo é **acíclico** se ele não contém circuitos.

**OBS:**: Os grafos desenhados no EXERCÍCIO A são todos acíclicos!

**Pergunta 3:** É verdade que se $G$ é um grafo conexo com $n$ vértices e $n-1$ arestas então $G$ é acíclico?
**Resposta 3:**

**Proposição 3.1.** Se $G$ é um grafo conexo com $n$ vértices e exatamente $n-1$ arestas então $G$ é acíclico.

**Prova** (na aula).

**Pergunta 4:** Vale a recíproca da Proposição 3.1?
**Resposta 4:**

**Proposição 3.2.** Se $G$ é um grafo conexo e acíclico com $n$ vértices, então $G$ tem exatamente $n-1$ arestas.

**Prova** (na aula).

Def. Uma **árvore** é um grafo conexo acíclico. Uma **floresta** é um grafo acíclico (não necessariamente conexo); ou seja, é um grafo cujos componentes são árvores.

Def. Numa árvore um vértice de grau 1 é chamado **folha**.

Juntando as Proposições 3.1 e 3.2, e a definição acima, temos:

**Teorema 3.3.** Um grafo conexo com $n$ vértices é uma árvore se e só se tem exatamente $n-1$ arestas.

**Corolário 3.4.** Toda árvore não trivial tem pelo menos 2 folhas.
**Prova** (na aula).

**Teorema 3.5.** As seguintes afirmações a respeito de um grafo $G$ são equivalentes:

(a) $G$ é uma árvore.

(b) $G$ não tem laços e entre quaisquer dois vértices de $G$ existe um único caminho.

(c) $G$ é acíclico e se $u$, $v$ são dois vértices não-adjacentes de $G$, então $G + uv$ tem exatamente um circuito. (Isto é, $G$ é um grafo acíclico maximal.)

(d) $G$ é conexo e se $e$ é uma aresta de $G$ então $G - e$ é desconexo. (Em outras palavras, $G$ é conexo e toda aresta de $G$ é uma ponte.)

**Prova.** [Na aula veremos algumas implicações. Complemente em casa.]

<u>Exercício 25.</u> *Prove que se $G$ é um grafo conexo com $n$ vértices e exatamente $n$ arestas, $n \geq 1$, então $G$ contém um único circuito.*

<u>Exercício 26.</u> *Prove que se $G$ é um grafo simples com pelo menos 5 vértices então ou $G$ ou o seu complemento $\bar{G}$ contém um circuito.*

<u>Def.</u> Uma **árvore geradora** (*spanning tree*) de um grafo $G$ é um subgrafo gerador de $G$ que é uma árvore. (Lembramos que um subgrafo $T$ de $G$ é gerador se $V(T) = V(G)$.)

Exemplos:

**Corolário 3.6.** Todo grafo conexo contém uma árvore geradora.

**Prova** (na aula).

Vértice-de-corte (ou vértice separador).

Num grafo $G$ um vértice $v$ é um **vértice-de-corte** se o conjunto das arestas de $G$ pode ser particionado em dois subconjuntos não-vazios $A_1$ e $A_2$ tais que os subgrafos $G[A_1]$ e $G[A_2]$ têm apenas o vértice $v$ em comum.

**OBS:** Se $G$ é um grafo *sem laços* então a definição acima é equivalente a: $v$ é um vértice-de-corte se $c(G - v) > c(G)$, isto é, o número de componentes de $G - v$ é maior do que o número de componentes de $G$.

Exemplos:

**Teorema 3.7.** Em uma árvore um vértice $v$ é um vértice-de-corte se e só se $g(v) > 1$.

**Prova** (na aula).

**Corolário 3.8.** Todo grafo conexo não-trivial, sem laços, tem pelo menos 2 vértices que não são vértices-de-corte.
**Prova** (na aula).

**Corolário 3.9.** Todo grafo conexo não-trivial $G$ tem um vértice $v$ tal que $G - v$ é conexo.

<u>Exercício 27.</u> *Seja $G$ um grafo. Prove que uma aresta $\alpha$ de $G$ é uma aresta-de-corte se e só se $\alpha$ não está contida em nenhum circuito de $G$.*

<u>Exercício 28.</u> *Seja $G$ um grafo conexo e $\alpha$ uma aresta de $G$. Prove que $\alpha$ pertence a todas as árvores geradoras de $G$ se e só se $\alpha$ é uma aresta-de-corte de $G$.*

**Teorema 3.10.** O número de árvores geradoras (rotuladas) distintas de $K_n$, $n \geq 2$, é igual $n^{n-2}$.

**Prova.** [Idéia a ser discutida em aula]

# APLICAÇÕES

## O PROBLEMA DA INTERLIGAÇÃO MÍNIMA

Deseja-se interligar (conectar) um certo número de locais através de uma rede de comunicação (fibra ótica). Sabendo-se que o custo para construir uma ligação direta de um local $i$ para um local $j$ é $c_{ij}$, deseja-se construir uma tal rede de forma que o custo total de construção seja o menor possível.

Considerando cada local $i$ como sendo um vértice $v_i$ de um grafo com custos $c_{ij}$ associados às arestas $v_i v_j$, o problema acima pode ser formulado da seguinte maneira:

Dado um grafo $G = (V, A)$, com custo $c(a) \geq 0$ associado a cada aresta $a \in A$, encontrar em $G$ um subgrafo gerador conexo de custo mínimo.

É imediato que no problema acima estamos interessados em encontrar em $G$ uma **árvore geradora de custo mínimo**. Uma tal árvore será chamada de **árvore ótima**.

O custo de um grafo $T$, denotado por $c(T)$, é definido como a soma dos custos das arestas em $T$; isto é

$$c(T) := \sum_{a \in T} c(a).$$

Exemplo:

## ALGORITMO DE KRUSKAL

Entrada: Grafo conexo $G = (V, A)$, com custo $c(a) \geq 0$ em cada aresta $a \in A$.
Saída: Árvore ótima $T$ (árvore geradora de custo mínimo).

1. (Ordenação) Ordene as arestas de $G$ em ordem não-decrescente de seus custos. Chame-as de $a_1, a_2, \ldots, a_m$, sendo $c(a_1) \leq c(a_2) \leq \ldots \leq c(a_m)$.
2. $F \leftarrow \emptyset$.
3. Para $i = 1$ até $m$ faça
   se $G[F \cup \{a_i\}]$ é acíclico então $F \leftarrow F \cup \{a_i\}$.
4. $T \leftarrow G[F]$. Pare.

**OBS:** O Passo 3 pode ser melhorado. Note que quando $|F| = |V| - 1$ não há mais necessidade de testar mais outras arestas.

**OBS:** Em aula, discussão sobre os casos em que os custos podem ser quaisquer (todos negativos; ou negativos e positivos).

O algoritmo acima é um *bom* algoritmo (muito eficiente). (Em aula: detalhes sobre a implementação e complexidade.)

**Teorema 3.11.** *Se $T$ é um subgrafo construído pelo algoritmo de Kruskal, então $T$ é uma árvore ótima de $G$.*

**Prova.** Seja $T$ um subgrafo construído pelo algoritmo de Kruskal. Claramente, $T$ é um subgrafo acíclico maximal de $G$ com $V(T) = V(G)$, e portanto, $T$ é uma árvore geradora de $G$. Vamos provar que $T$ é uma árvore ótima.

Suponha que $A(T) = \{e_1, e_2, \ldots, e_k\}$, e que $e_i$ foi escolhido antes de $e_j$ se $i < j$.

Escolha uma árvore ótima $T^*$ de $G$ tal que $T^*$ tenha o maior número possível de arestas em comum com $T$ (isto é, tal que $|A(T^*) \cap A(T)|$ seja máxima). Vamos provar que $T^* = T$.

Suponha que $T^* \neq T$. Seja $e_j$ a primeira aresta escolhida para pertencer a $T$ que não é uma aresta de $T^*$. (Isto significa que as arestas $e_1, \ldots, e_{j-1}$ pertencem a $T$ e a $T^*$). Sejam $u$ e $v$ os extremos da aresta $e_j$, e seja $P$ o (único) caminho em $T^*$ que vai de $u$ para $v$. Note que o caminho $P$ tem pelo menos uma aresta, digamos $xy$, que não pertence a $T$ (caso contrário, $T$ conteria um circuito). Como a aresta $uv$ ($= e_j$) foi escolhida pelo algorimto e $xy$ não foi, segue que $c(xy) \geq c(uv)$. De fato, basta notar que $G[\{e_1, \ldots, e_{j-1}\} \cup \{xy\}]$ é acíclico, e portanto, se tivéssemos $c(xy) < c(uv)$, então a aresta $xy$ deveria ter sido escolhida pelo algoritmo.

Seja $T' := T^* + uv - xy$; Note que $T'$ é uma árvore geradora [justifique esta afirmação] e $c(T') = c(T^*) + c(uv) - c(xy) \leq c(T^*)$. Como $T^*$ é uma árvore ótima, então $c(T') = c(T^*)$, e portanto $T'$ é também uma árvore ótima. Mas $T'$ tem mais arestas em comum com $T$ do que $T^*$ (isto é, $|A(T') \cap A(T)| > |A(T^*) \cap A(T)|$), uma contradição à escolha de $T^*$. Logo, devemos ter $T^* = T$, o que completa a prova. ∎