

# ***ESTRUTURAS DE DADOS***

## 1. GENERALIDADES

### - Tipos de Estruturas de Dados

- matriz
- cadeia de caracteres
- lista linear
- pilha
- fila
- árvore
- grafo

### - Operações que podem ser feitas numa estrutura de dados:

- inserção
- remoção
- leitura
- substituição

### - Computadores servem para armazenar informações e programas para manipulá-las.

Programa é um algoritmo que manipula dados.

Estrutura de dados retrata as relações lógicas existentes entre os dados.

## 2. MATRIZ

É uma estrutura de dados onde a informação armazenada (nó) tem a sua identificação feita através de uma linha e uma coluna.

Ex.:

$$M = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix}$$

Aplicações de Matrizes:

- cálculo de potências num sistema de potência

## 3. CADEIA DE CARACTERES

O tipo de dados primitivo caracter consiste de um só caracter. No entanto, na prática, muitas vezes tem-se de manipular dados tais como nomes, frases e textos, que envolvem mais de um caracter e que são denominados CADEIA DE CARACTERES.

## 4. LISTA LINEAR

### 4.1 Introdução

É uma estrutura de dados que tem uma sequência ordenada de dados denominados nós.

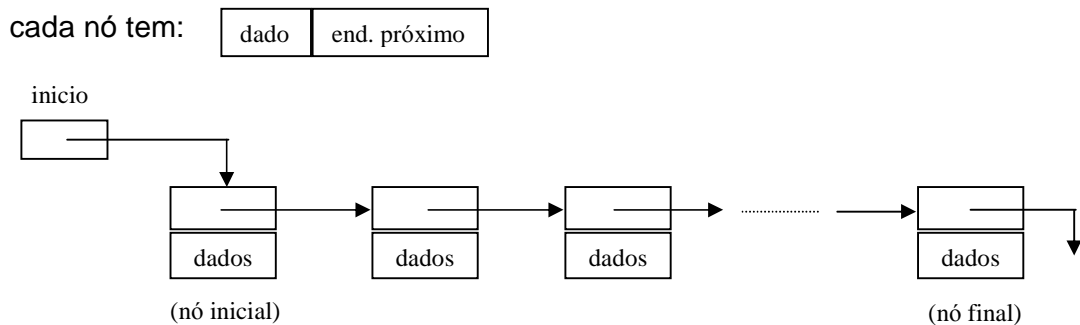
Cada Lista Linear tem o seu nó inicial e o seu nó final, onde cada nó apresenta o mesmo tipo de dado.

Cada nó possui um campo de dados onde se armazenam as informações e um campo de ponteiros onde se armazenam os endereços que implementam as conexões entre os nós.

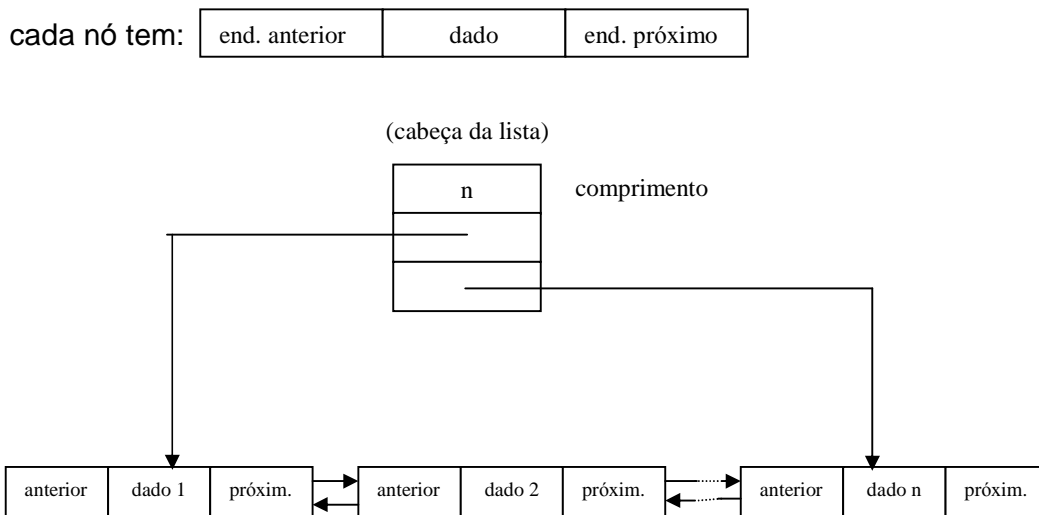
Existem vários tipos de Listas Lineares:

- lista ligada simples
- lista duplamente ligada

### 4.2 Lista ligada simples



### 4.3 Lista duplamente ligada



### 4.4 Exemplos de Lista Linear

- dias da semana
- cartas de baralho
- andares de um edificio

## 5. PILHA

É uma estrutura linear LIFO ("Last In First Out") (Primeiro a Entrar Último a Sair)

Ex.: pilha de livros

É uma Lista Linear na qual todas as inserções e retiradas são feitas na mesma extremidade (o topo).

## 6. FILA

É uma estrutura linear FIFO ("First In First Out") (Primeiro a Entrar Primeiro a Sair).

Ex.: fila de pessoas em um guichê do banco

É uma Lista Linear na qual todas as inserções ocorrem numa extremidade (a traseira) e as retiradas ocorrem na outra extremidade (a frente)

## 7. ÁRVORE

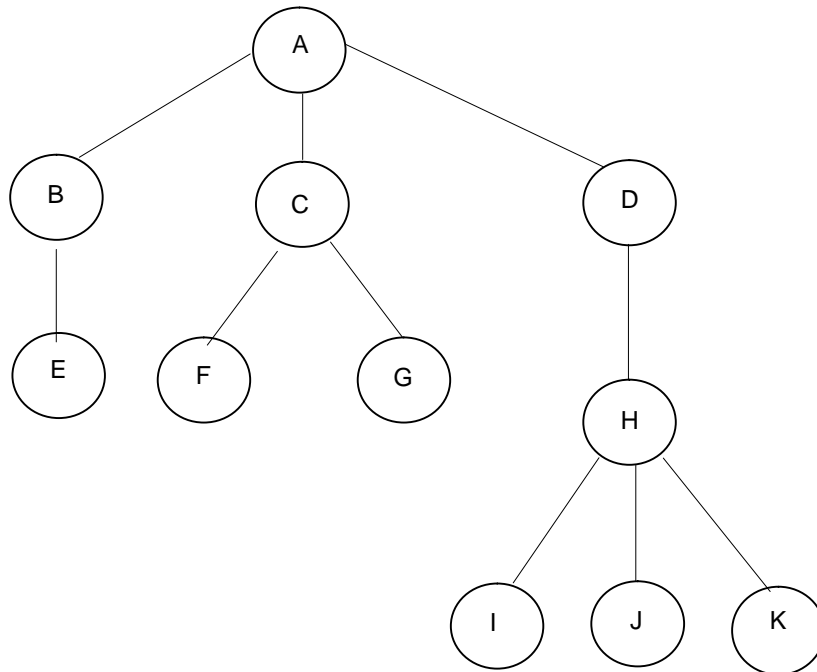
### 7.1 Introdução

Na Árvore a relação existente entre os dados (denominados nós) é uma relação de hierarquia ou de composição, onde um conjunto de dados é hierarquicamente subordinado a outro.

Uma Árvore é um conjunto finito  $T$  de um ou mais nós, tais que:

- a) existe um nó raiz da árvore
- b) os demais nós formam conjuntos disjuntos, onde cada um destes conjuntos é uma árvore (denominada de subárvores)

Exemplo:



### Terminologia

- o número de subárvores de um nó é o grau daquele nó;
- cada nó da Árvore é a raiz de uma subárvore;
- um nó de grau zero é denominado folha ou nó terminal;
- nível do nó: é igual ao número de "linhas" que o liga à raiz (a raiz tem nível zero);
- altura da Árvore é definida como sendo o nível mais alto da Árvore;
- floresta é um conjunto de zero ou mais Árvores disjuntas (se eliminarmos o nó raiz de uma árvore, o que dela restar forma uma floresta);
- a Árvore é ordenada quando a ordem dos subárvores é significativa;
- a Árvore é orientada quando a ordem das subárvores não é relevante.

Ex.:



para o caso de Árvore orientada, as duas árvores são iguais

- a raiz de uma árvore é chamada de pai das raízes das suas subárvores

- as raízes das subárvores de um nó são chamadas de irmãos (que são filhos de seu nó pai)
- Árvores Binárias são estruturas do tipo Árvore, onde o grau de cada nó é menor ou igual a dois.

Para o exemplo temos:

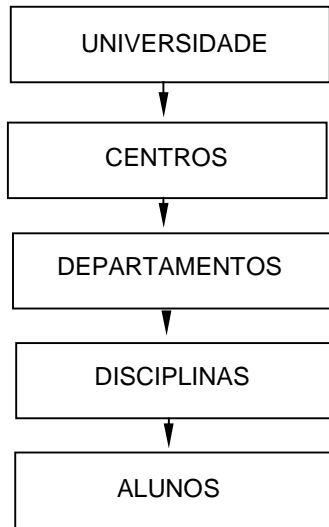
N	GRAU	NÍVEL
A	3	0
B	1	1
C	2	1
D	1	1
E	0	2
F	0	2
G	0	2
H	3	2
I	0	3
J	0	3
K	0	3

Exemplos de Árvore:

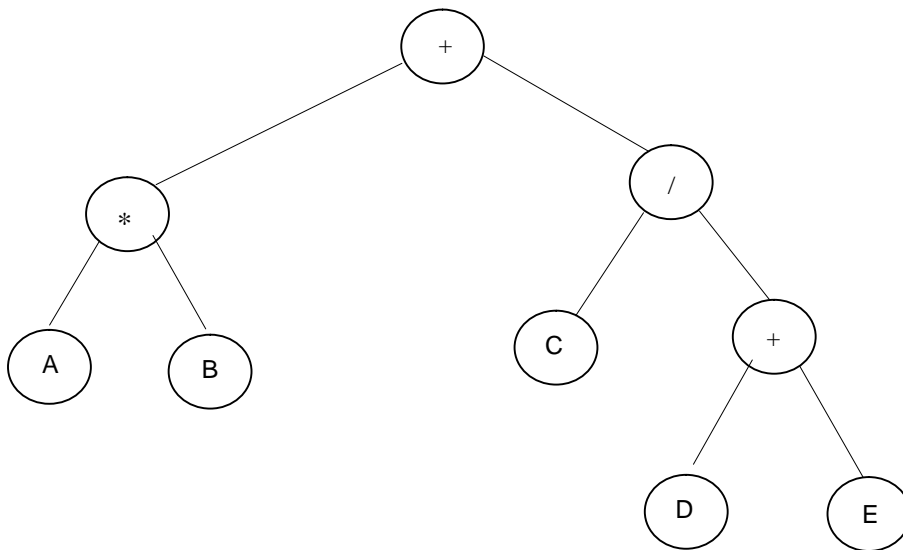
- árvore genealógica de uma família
- organograma de uma empresa
- índice de um livro
- diretórios e arquivos num sistema de computação

## 7.2 Aplicação de Árvores

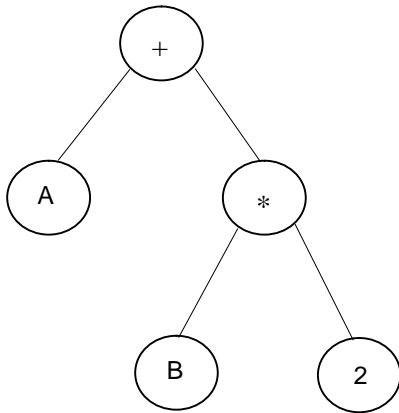
- Esquema hierárquico de uma universidade



- Equação  $\rightarrow A * B + C / (D + E)$

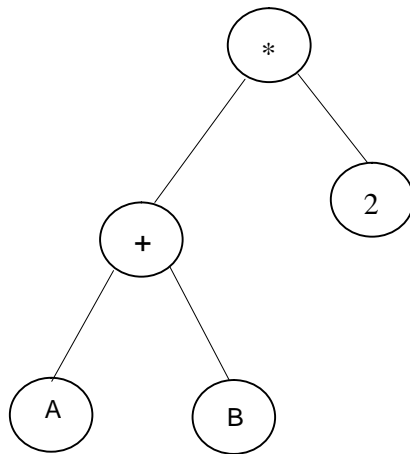


- Equação  $\rightarrow A + B * 2$



o operador de menor prioridade da expressão aritmética aparece na raiz da Árvore

- Equação  $(A + B) * 2$





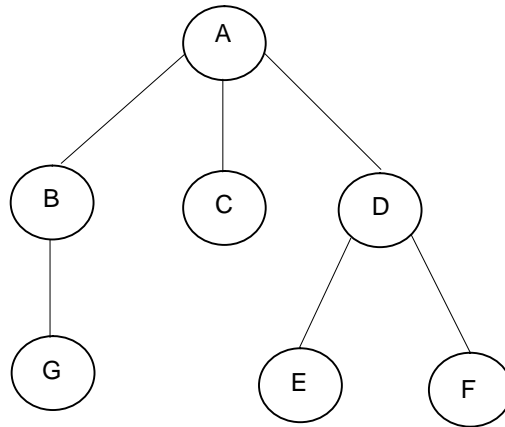
### 7.3 Alocação das Árvores na Memória do Computador

- por adjacência
- por encadeamento

#### a) Alocação por Adjacência

- os nós da Árvore são representados seqüencialmente na memória, de acordo com uma determinada ordem convencionada em que eles aparecem na Árvore;
- os números que aparecem à direita da informação de cada nó na alocação seqüencial indicam o grau do nó correspondente.

Ex.:



A	3	B	1	G	0	C	0	D	2	E	0	F	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---

representação na memória

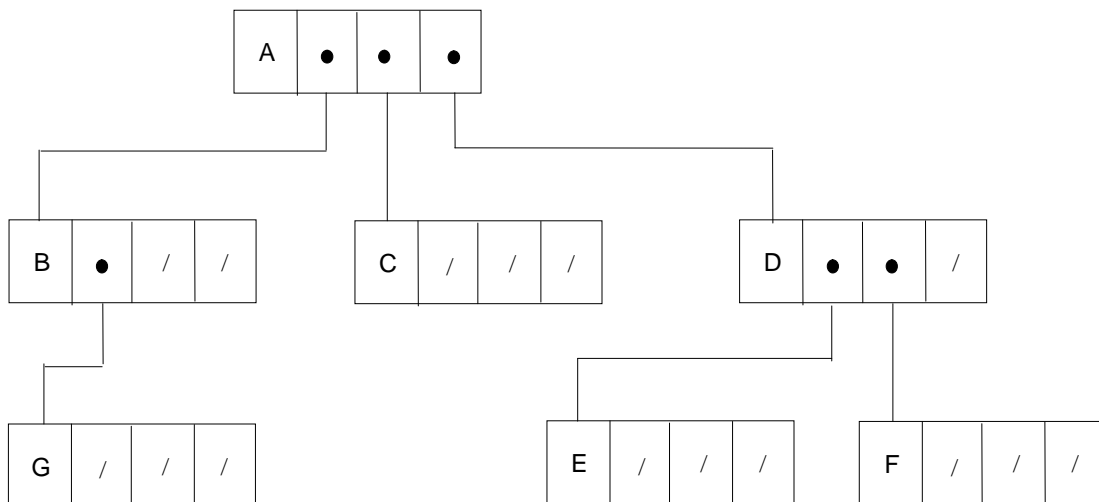
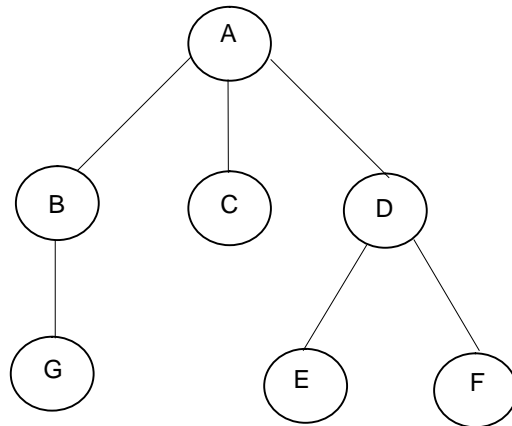
Obs.:

- a alocação seqüencial não constitui na maioria dos casos uma maneira conveniente para representar Árvores (dificuldades para inserções, remoções e localizações de um nó particular)
- essa forma de alocação é útil para armazenamento permanente (em fita ou disco magnético) de Árvores representadas por encadeamento

## b) Alocação por Encadeamento

- cada nó é um dado que pode ser alocado dinamicamente e possui espaço para representar tanto a informação do nó como as referências das subárvores daquele nó;
- o tipo de dado dos nós tem tantas referências quanto é o número máximo de subárvores que um nó possa ter.

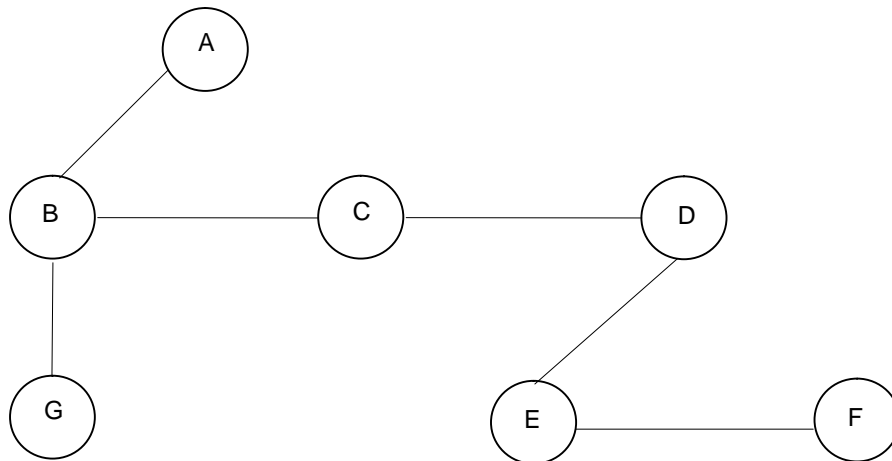
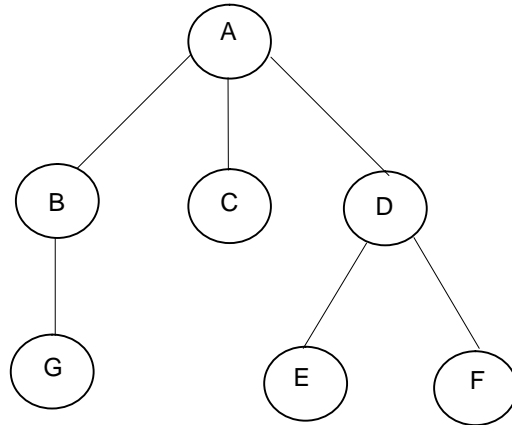
Ex.:



## 7.4 Transformação de Árvore qualquer para Árvore Binária

- Árvore Binária -> grau máximo é 2
- para se fazer a transformação deve-se ligar os nós irmãos e remover a ligação entre um nó pai e os nós filhos, exceto os do primeiro filho.

Ex.:



- para se interpretar corretamente a hierarquia de uma Árvore transformada em Árvore Binária deve-se observar que a subárvore da esquerda de um nó é o filho deste nó, enquanto que a subárvore da direita é seu irmão.

## 8. GRAFO

Um Grafo  $G$  é constituído por um conjunto  $N$  de elementos e por uma relação binária  $A$  entre esses elementos.

$$G = (N, A)$$

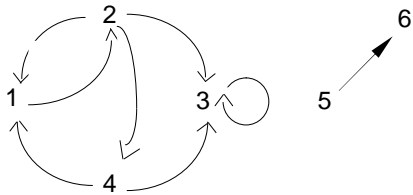
$A =$  conjunto cujos membros são pares ordenados  $(n_i, n_j)$ , onde  $n_i$  e  $n_j$  são elementos de  $N$ .

Os elementos de  $N$  são denominados nós (ou vértices), enquanto os elementos de  $A$  são denominados arcos (ou arestas).

Exemplo:

$$N = \{1, 2, 3, 4, 5, 6\}$$

$$A = \{(1,2), (2,1), (2,3), (2,4), (3,3), (4,1), (4,3), (5,6)\}$$



Nós adjacentes são nós ligados por arcos. Ex.: o nó 2 é adjacente a 1, 3 e 4.

Caminho = sequência de um ou mais arcos

$$\langle (a, n_1), (n_1, n_2), \dots, (n_{i-1}, n_i), (n_i, b) \rangle$$

Quando  $a = b$  temos um circuito.

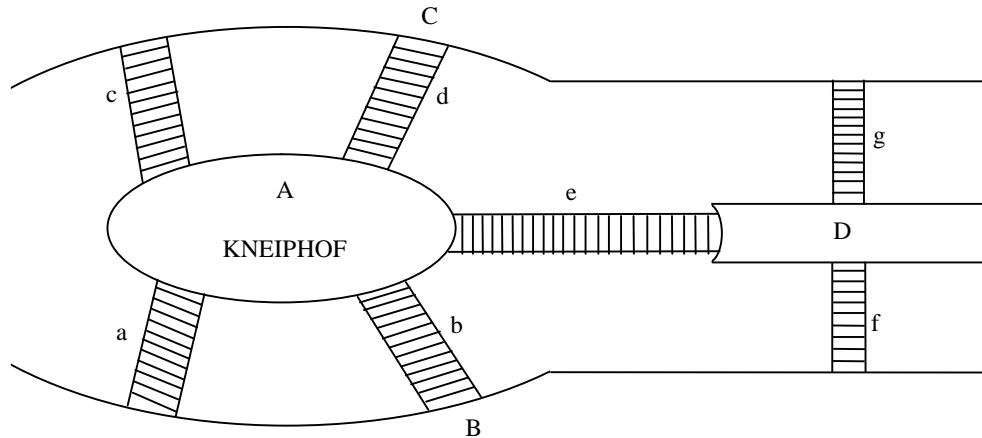
Grafo é conexo quando tem um nó do qual existem caminhos para todos os demais.

Subgrafo = subconjunto dos nós de um dado grafo

1ª Aplicação de Grafos:

1736

EULER → PONTE DE KOENIGSBERG

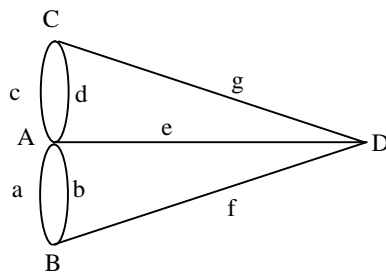


PROBLEMA

sair de uma área de terra, passar por todas pontes somente uma vez, e voltar a área de terra inicial.

EULER provou que era impossível, sendo:

- áreas de terra = vértices
- pontes = bordas



grau de um vértice = número de bordas que lhe são incidentes

EULER mostrou que para existir um caminho é necessário que seja de valor par o grau de cada vertice.

Um grafo G consiste de 2 conjuntos V e E.

- V = conjunto de vértices
- E = conjunto de bordas

Aplicações de Grafos:

- análise de circuitos elétricos
- verificação de caminhos mais curtos
- análise de planejamento de projetos
- identificação de produtos químicos