

Algoritmos e Estrutura de Dados II

Prof. Fellipe Guilherme Rey de Souza

Aula 08 – Árvore

Agenda

O que é uma Árvore?

Árvore – Definições

Árvore – Tipos

Árvore – Altura

Árvore – Representação

→ **Árvore – Percurso**

- O que é uma Árvore?
- Árvore Binária
 - Definições
 - Tipos
 - Altura
 - Representação
 - Percurso

O que é uma Árvore?

O que é uma Árvore?

→ O que é uma Árvore?

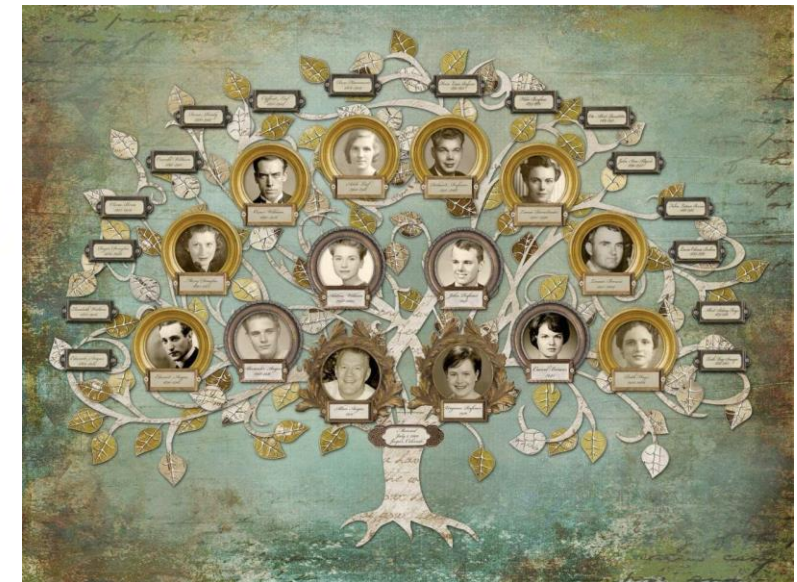
Árvore – Definições

Árvore – Tipos

Árvore – Altura

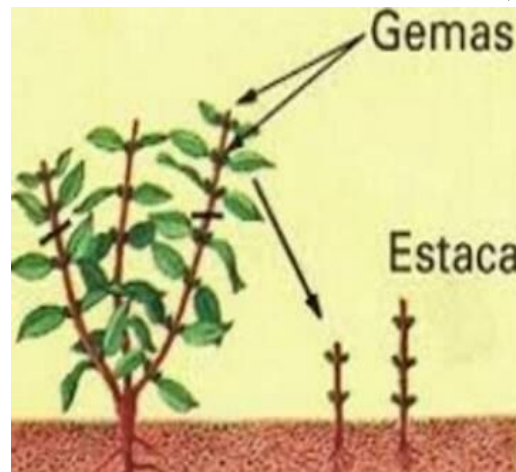
Árvore – Representação

Árvore – Percurso



O que é uma Árvore?

- O termo **Árvore** nos remete a uma estrutura hierárquica em que, saindo de uma raiz, podemos ter vários nós filhos (também chamados de descendentes).
 - Cada nó filho também pode ter outros filhos, já que ele pode ser a raiz de uma subárvore.



O que é uma Árvore?

- Na computação, o termo **árvore** é uma estrutura de dados hierárquica composta por nós, onde cada nó possui um valor e uma lista de referências (ou ponteiros) para outros nós, chamados de **filhos**.
- A estrutura é organizada de maneira que um nó pode ter um nó "pai" e um ou mais nós "filhos", mas cada nó, exceto o nó raiz, tem exatamente um nó pai.

Árvore – Definições

- São os componentes de uma árvore:
 - i. **Raiz (Root):** É o nó principal da árvore, de onde todos os outros nós se ramificam. A árvore sempre tem uma única raiz.
 - ii. **Nós (Nodes):** Cada elemento da árvore. Um nó pode ter um valor e um ponteiro (ou lista de ponteiros) para seus filhos.
 - iii. **Filhos (Children):** Nós que são descendentes diretos de outro nó. Cada nó pode ter múltiplos filhos.
 - iv. **Pai (Parent):** O nó diretamente acima de um nó em relação à hierarquia.

Árvore – Definições

- São os componentes de uma árvore (*cont.*):
 - v. **Folhas (Leaves)**: Nós que não têm filhos, ou seja, são os nós mais baixos da árvore.
 - vi. **Subárvore (Subtree)**: Qualquer nó de uma árvore, junto com todos os seus filhos e descendentes, pode ser considerado uma subárvore.
 - vii. **Nível (Level)**: A profundidade de um nó na árvore, onde o nó raiz está no nível 0, seus filhos estão no nível 1, e assim por diante.
 - viii. **Altura (Height)**: A altura de uma árvore é a maior distância entre a raiz e uma folha.

Árvore – Definições

O que é uma Árvore?

→ **Árvore – Definições**

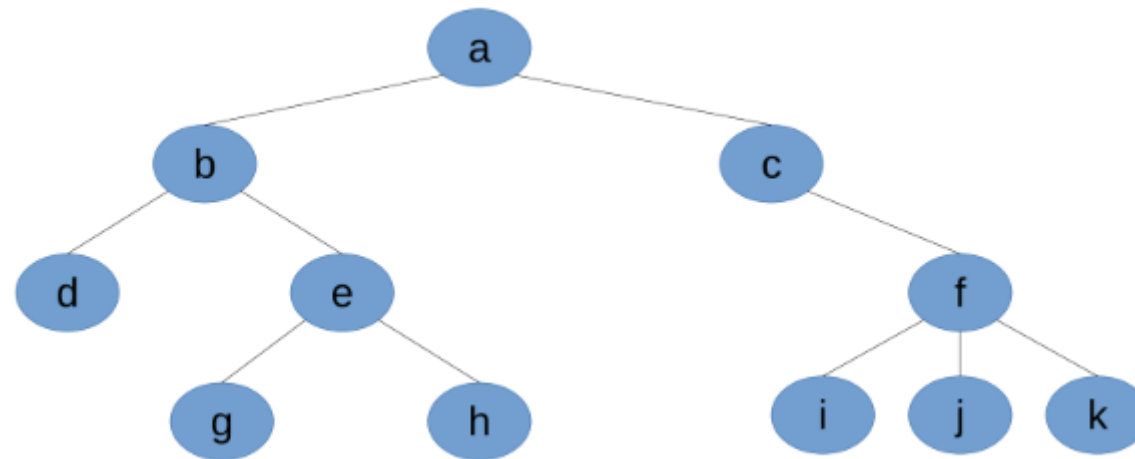
Árvore – Tipos

Árvore – Altura

Árvore – Representação

Árvore – Percurso

- Seja a seguinte árvore:



PS: Exemplos a partir de agora retirados do material do Prof. Flávio Barbieri Gonzaga.

Árvore – Definições

O que é uma Árvore?

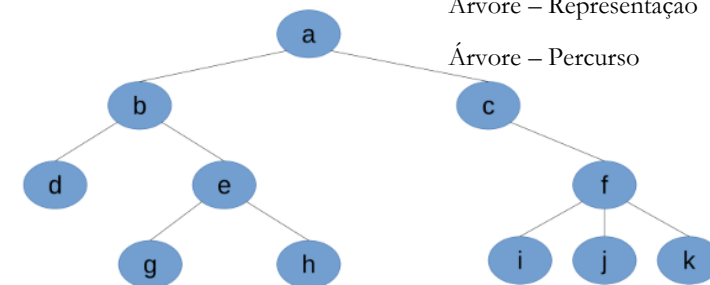
→ **Árvore – Definições**

Árvore – Tipos

Árvore – Altura

Árvore – Representação

Árvore – Percurso



- Temos as seguintes definições:
 - i. **Raiz:** nó **a**, mas todo nó é raiz de uma subárvore.
 - ii. **Grau:** número de subárvores do nó, logo, **a** tem grau 2.
 - iii. **Nó de grau 0:** nó folha ou terminal. Ex: nó **d, g, h, i, j, k**.
 - iv. **Nível de um nó:** A raiz de uma árvore tem nível 1. Se um nó tem nível i , seus filhos terão nível $i+1$. Ex: nó **e** tem nível 3.
 - v. **Altura:** é o máximo nível de seus nós. A árvore acima tem altura 4.

Árvore – Definições

O que é uma Árvore?

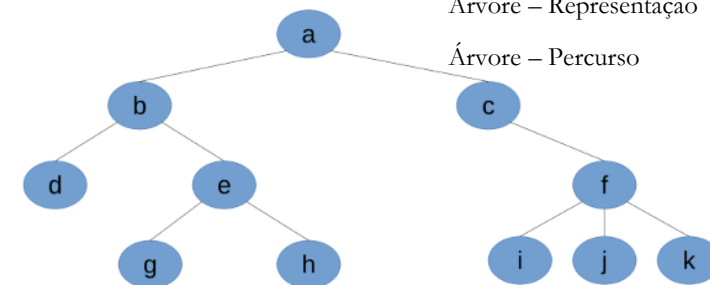
→ Árvore – Definições

Árvore – Tipos

Árvore – Altura

Árvore – Representação

Árvore – Percurso



- Temos as seguintes definições (*cont.*):

vi. **Pai e filho:** o nó **f**, por estar abaixo de **c**, e diretamente ligado a ele, é chamado de filho de **c**. De forma semelhante, **c** é dito pai de **f**.

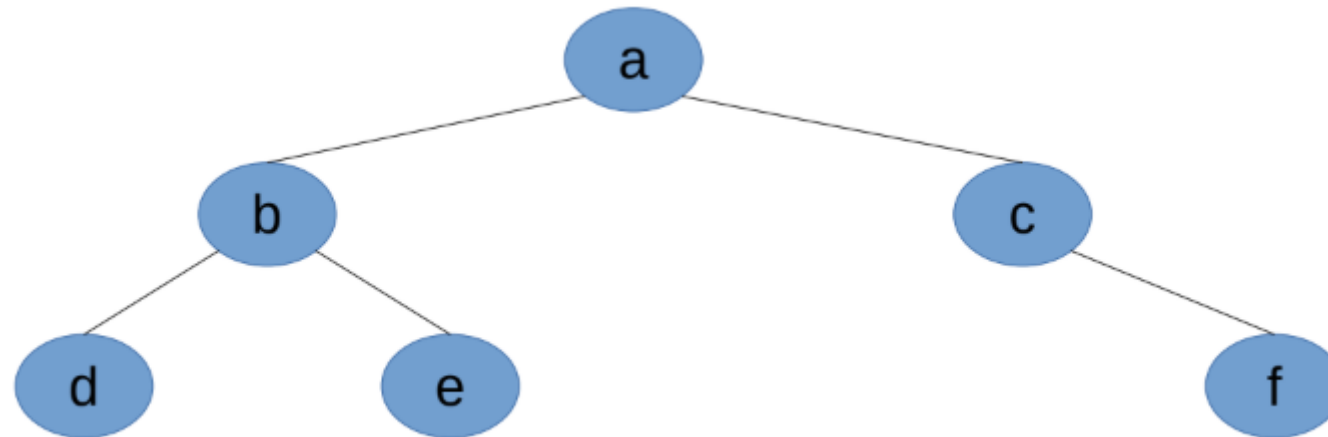
vii. **Irmão:** nós com o mesmo pai são ditos irmãos. Ex: **g** e **h**.

viii. **Nó interno:** nó que não é folha.

ix. **Floresta:** conjunto de 2 ou mais árvores.

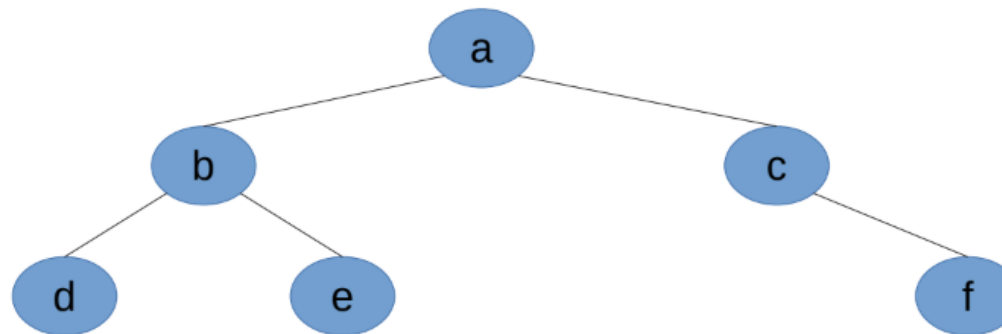
Árvore Binária

- É chamada de **Árvore Binária** àquelas onde o grau de cada nó é menor ou igual a dois.
 - Ou seja, cada nó pode ter, no máximo, dois filhos.



Árvore Binária

- É chamada de **Árvore Binária** àquelas onde o grau de cada nó é menor ou igual a dois (*cont.*).
 - Existe a distinção entre subárvore direita e esquerda. Para esse exemplo, a subárvore direita de **a** começa em **c**, e a subárvore esquerda de **a** começa em **b**.



Árvore Binária – Tipos

- Alguns tipos especiais de árvores binárias são:
 - Estritamente binária: cada nó possui 0 ou 2 filhos.
 - Completa: se v é um nó tal que alguma subárvore de v é vazia, então v se localiza ou no último ou no penúltimo nível da árvore.
 - Cheia: se v é um nó com alguma das sua subárvores vazias, então v se localiza no último nível.

Árvore Binária – Tipos

O que é uma Árvore?

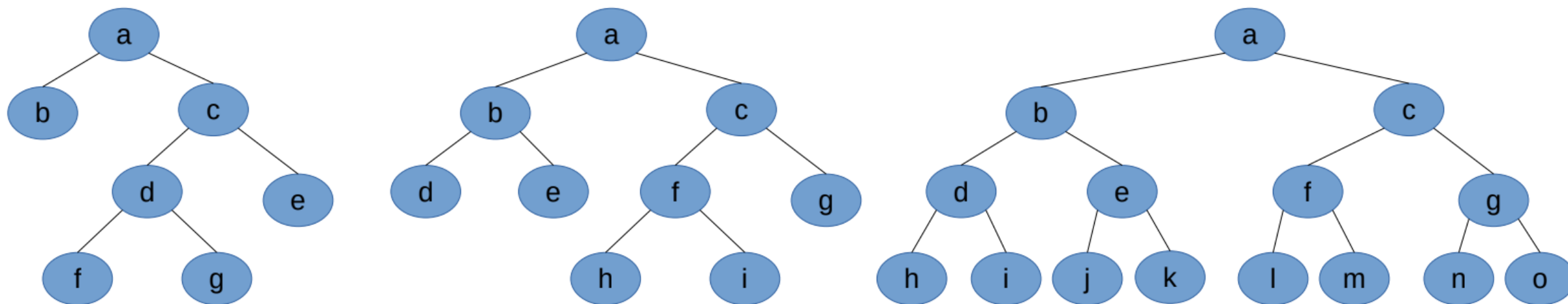
Árvore – Definições

→ **Árvore – Tipos**

Árvore – Altura

Árvore – Representação

Árvore – Percurso



Estritamente binária: cada nó possui 0 ou 2 filhos.

Completa: se v é um nó tal que alguma subárvore de v é vazia, então v se localiza ou no último ou no penúltimo nível da árvore.

Cheia: se v é um nó com alguma das suas subárvores vazias, então v se localiza no último nível.

Árvore Binária – Altura

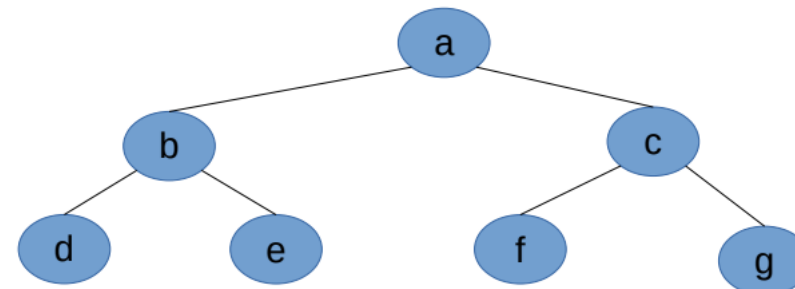
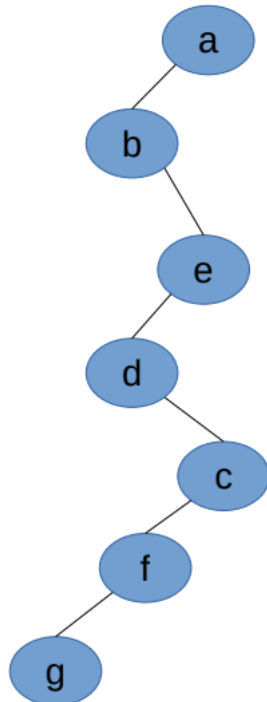
- A relação entre a altura de uma árvore binária e seu número de nós é um dado importante para várias aplicações.
- Para um valor fixo de quantidade de nós (**n**), quais são as árvores que possuem altura (**h**) máxima e mínima?

Árvore Binária – Altura

- Altura máxima será aquela cujos nós anteriores possuem exatamente uma subárvore vazia (denominadas zigue-zague).
- Uma árvore completa apresenta a altura mínima.

Árvore Binária – Altura

- Para um valor fixo de quantidade de nós (**n**), quais são as árvores que possuem altura (**h**) máxima e mínima?



Árvore Binária – Representação

- O armazenamento de uma árvore binária surge naturalmente de sua definição.
 - Cada nó deve possuir dois campos de ponteiros, **esq** e **dir**, que apontam para suas subárvores esquerda e direita respectivamente.
 - O ponteiro **ptrai** indica a raiz da árvore.
 - Os campos do nó da árvore que contêm as informações relacionadas ao problema serão representados como um só campo de nome **info**.

Árvore Binária – Representação

O que é uma Árvore?

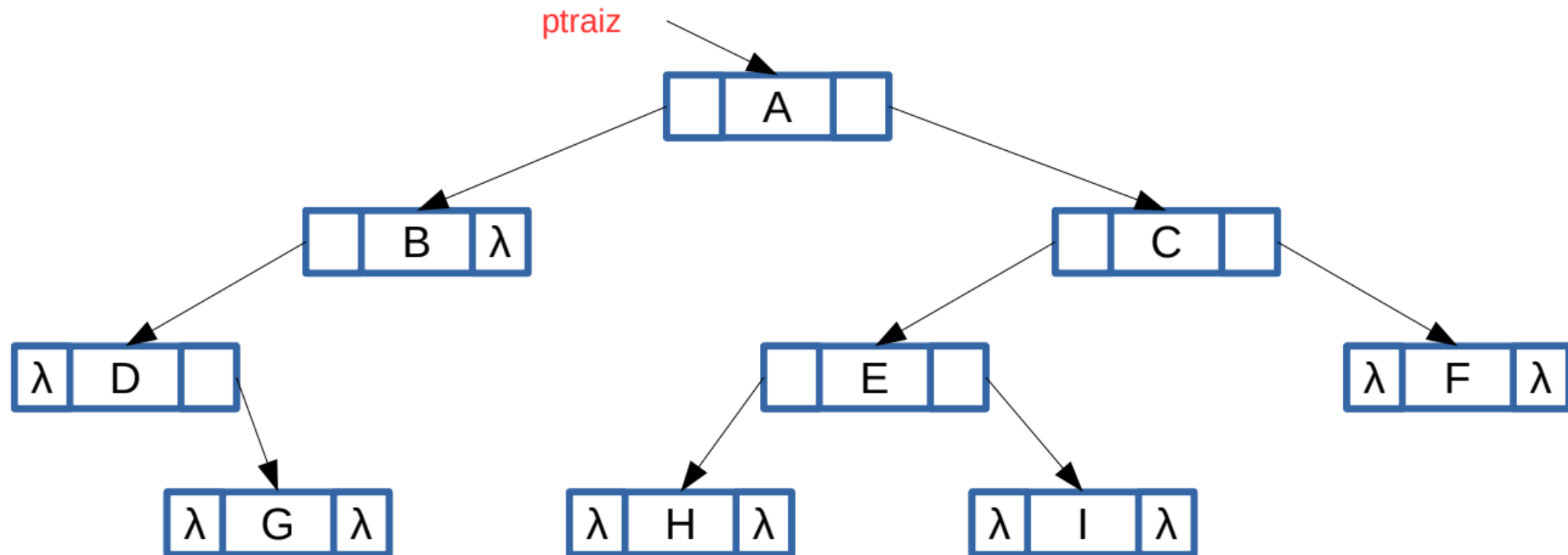
Árvore – Definições

Árvore – Tipos

Árvore – Altura

→ **Árvore – Representação**

Árvore – Percurso



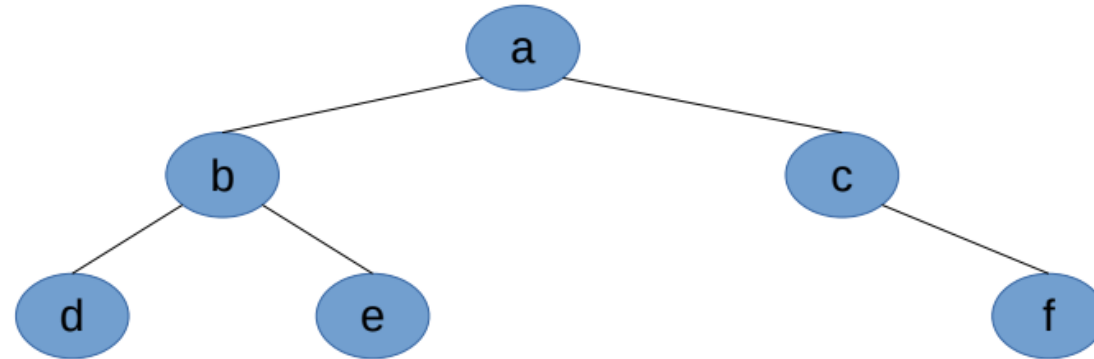
$\lambda = \text{NULO}$

Árvore Binária – Percurso

- Caminhar em uma árvore binária significa percorrer todos os nós de forma sistemática, de modo que cada nó seja visitado uma vez.
 - Visitar um nó significa operar, de alguma forma, com a informação à ele relativa.
 - No processo de percorrer uma árvore pode ser necessário passar várias vezes por alguns de seus nós, sem visitá-los.

Árvore Binária – Percurso

- Usando-se de recursividade, existem 3 formas básicas de caminhamento em árvore binária:



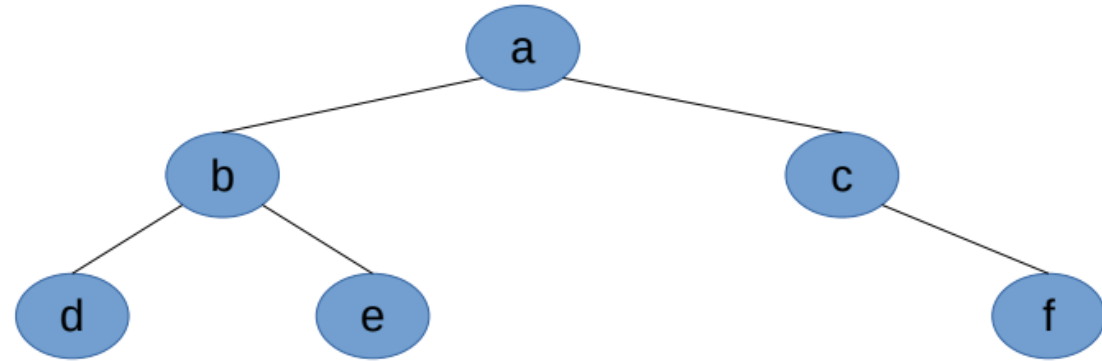
- **Pré-ordem:**

1. Processa o nó.
2. Percorre recursivamente, em pré-ordem, a subárvore esquerda.
3. Percorre recursivamente, em pré-ordem, a subárvore direita.

Pré-ordem: a, b, d, e, c, f

Árvore Binária – Percurso

- Usando-se de recursividade, existem 3 formas básicas de caminhamento em árvore binária:



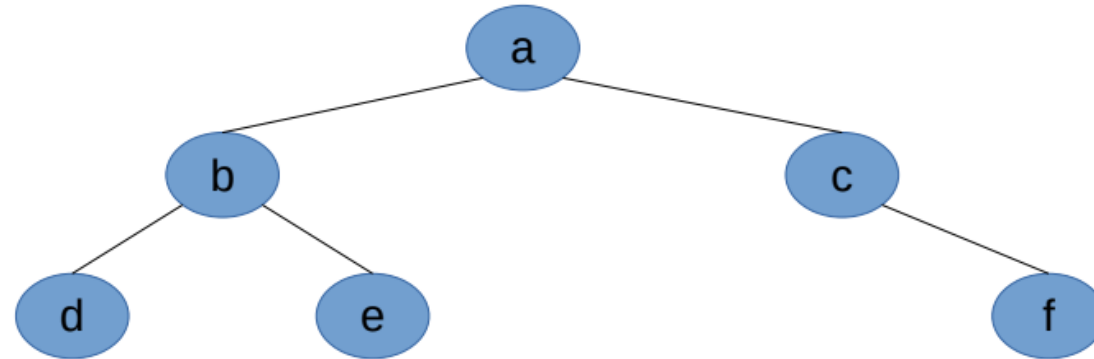
- **Em-ordem:**

1. Percorre recursivamente, em ordem, a subárvore esquerda.
2. Processa o nó.
3. Percorre recursivamente, em ordem, a subárvore direita.

Em-ordem: d, b, e, a, c, f

Árvore Binária – Percurso

- Usando-se de recursividade, existem 3 formas básicas de caminhamento em árvore binária:



- **Pós-ordem:**

1. Percorre recursivamente, em pós-ordem, a subárvore esquerda.
2. Percorre recursivamente, em pós-ordem, a subárvore direita.
3. Processa o nó.

Pós-ordem: d, e, b, f, c, a

Árvore Binária – Percurso

```
pre-ordem(pt){  
  visita(pt);  
  se pt^.esq  $\neq$   $\lambda$  então pre-ordem(pt^.esq)  
  se pt^.dir  $\neq$   $\lambda$  então pre-ordem(pt^.dir)  
}
```

```
em-ordem(pt){  
  se pt^.esq  $\neq$   $\lambda$  então em-ordem(pt^.esq)  
  visita(pt);  
  se pt^.dir  $\neq$   $\lambda$  então em-ordem(pt^.dir)  
}
```

```
pos-ordem(pt){  
  se pt^.esq  $\neq$   $\lambda$  então pos-ordem(pt^.esq)  
  se pt^.dir  $\neq$   $\lambda$  então pos-ordem(pt^.dir)  
  visita(pt);  
}
```

```
visita(pt){  
  imprime(pt^.info);  
}
```

Árvore Binária – Percurso

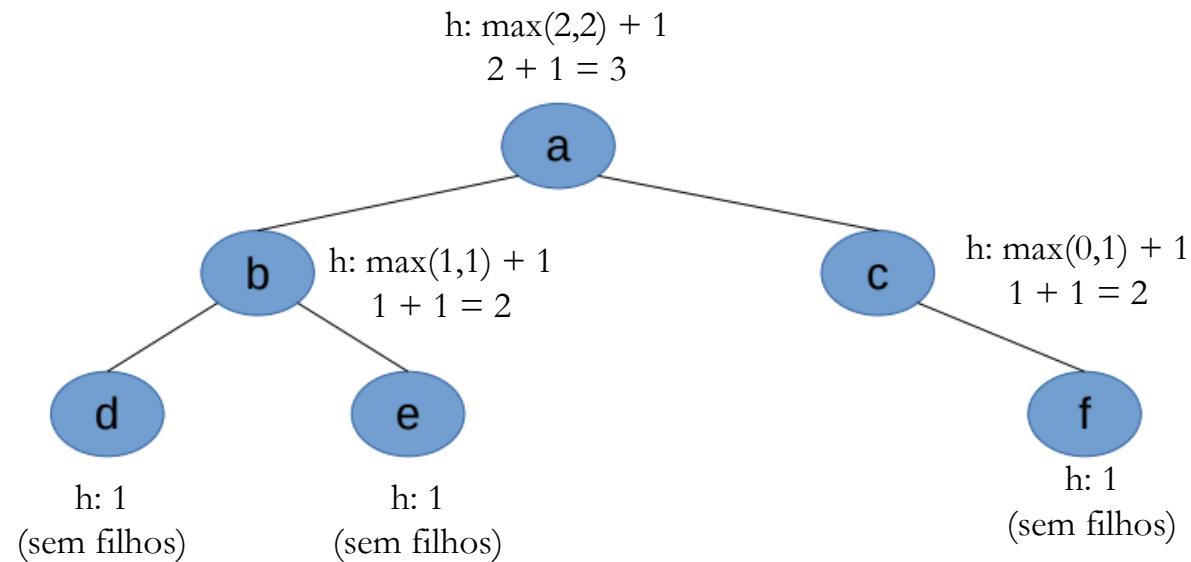
- Qual dos métodos de caminhamento que, ao ser usado juntamente com a função visita, produziria o resultado de calcular a altura do nó?
 - O caminhamento mais comum utilizado é o **caminhamento em profundidade** (caminhamento pós-ordem / *post-order traversal*).
 - O caminhamento pós-ordem visita um nó após visitar seus filhos esquerdo e direito, o que é útil para calcular a altura da árvore porque, para determinar a altura de um nó, você precisa conhecer as alturas de seus filhos antes de calcular a altura do nó em si.

Árvore Binária – Percurso

- Qual dos métodos de caminhamento que, ao ser usado juntamente com a função visita, produziria o resultado de calcular a altura do nó? (*cont.*)
 - Podemos resumir no seguinte algoritmo:
 - i. Para cada nó, calcule a altura de seus filhos (esquerdo e direito).
 - ii. A altura do nó é a maior altura entre seus filhos, mais 1.
 - iii. Continue o processo recursivamente até a raiz.

Árvore Binária – Percurso

- Qual dos métodos de caminhamento que, ao ser usado juntamente com a função visita, produziria o resultado de calcular a altura do nó? (*cont.*)



Pós-ordem: d, e, b, f, c, a

Operações em Árvore

- O TAD Árvore possui algumas operações básicas, que são:
 - **Inserir (*insert*)**: Adicionar um novo elemento na árvore.
 - **Remover (*remove*)**: Remover um elemento da árvore.
 - **Busca (*search*)**: Verificar (sem remover) se existe o item na árvore.
 - **Percurso (*tree traversal*)**: Percorrer a árvore
 - **Tamanho (*size*)**: Verifica o tamanho da árvore.

Algoritmos e Estrutura de Dados II

Prof. Fellipe Guilherme Rey de Souza

Aula 08 – Árvore