



IPS Instituto
Politécnico de Setúbal
**Escola Superior de
Tecnologia de Setúbal**

Programação Avançada 2021-22

[2a] Árvores | Estruturas de dados ~ Conceitos

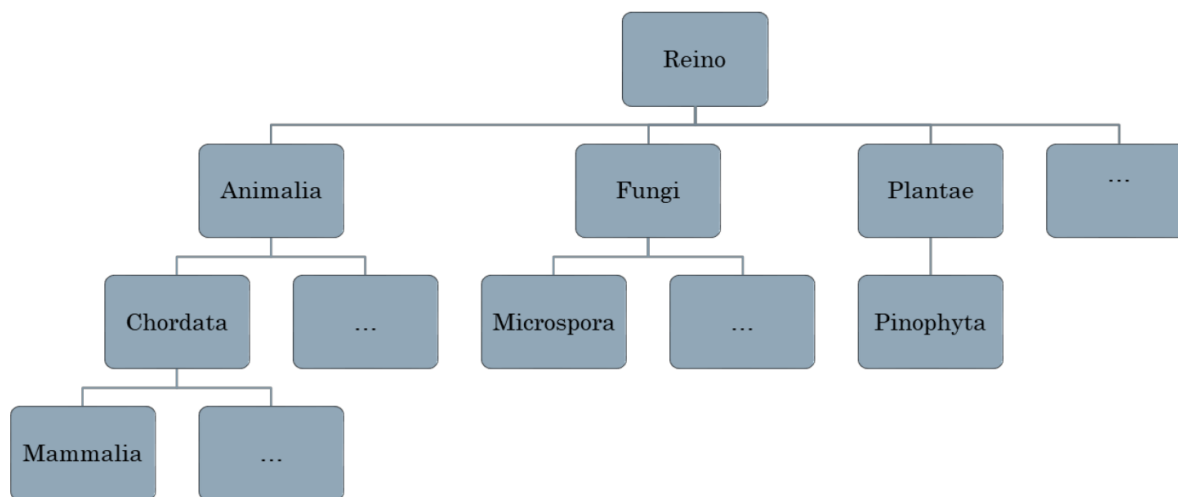
Bruno Silva, Patrícia Macedo

Sumário

- Árvores como estruturas de dados hierárquicas
- Conceitos
 - Grau, Ordem, Níveis, Altura e subárvores
 - Travessia (inglês: *traversal*) de árvores
 - *breadth-first* e *depth-first*
 - Árvores binárias
- Árvores e algoritmos recursivos
- Exercícios

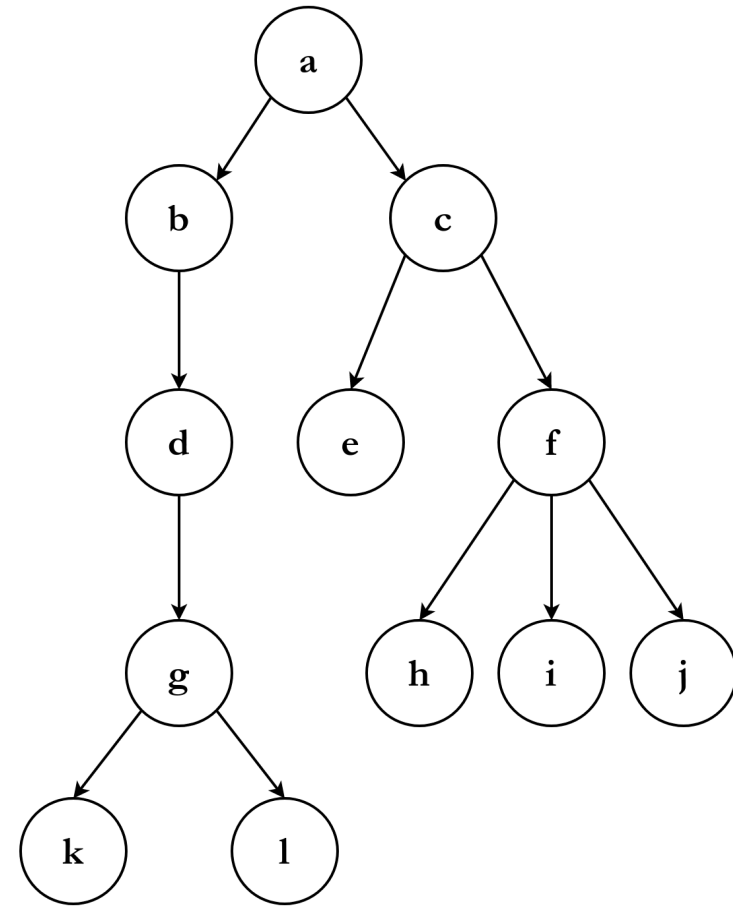
Árvores

- As **árvores**, no contexto das ciências da computação, são estruturas de dados não-lineares e hierárquicas.
- Permitem representar (informação de) elementos com relações hierárquicas, i.e., relações de *pai, filho, ascendente e descendente*. Exemplo:



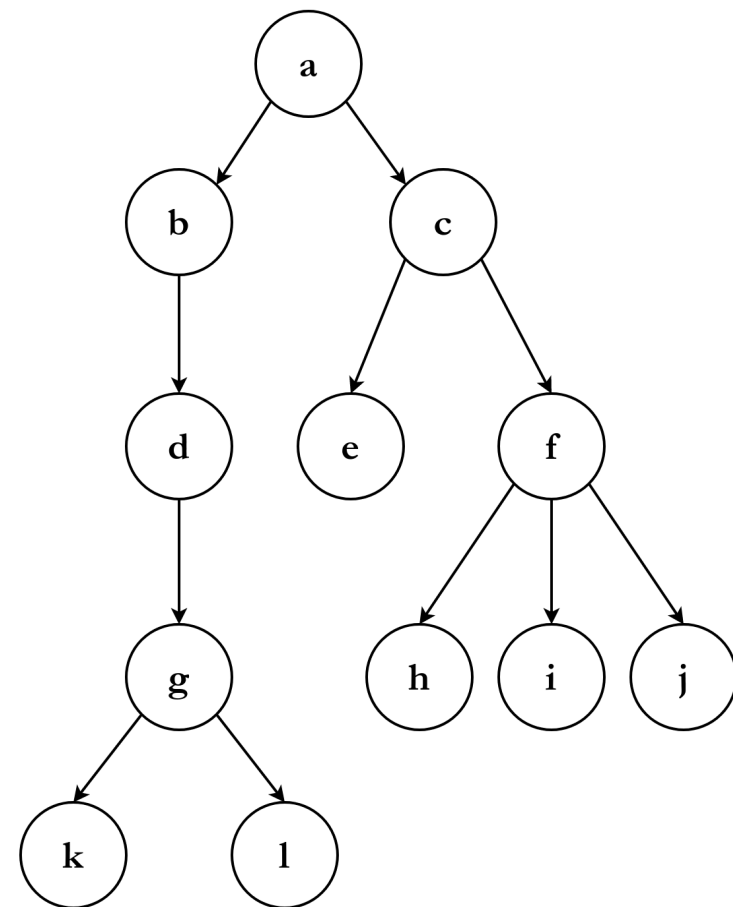
Árvores | Conceitos

- Uma árvore é composta por **nós**;
- No topo da árvore existe um nó especial, a **raiz**; não possui ascendentes - nó **a**.
- Dos restantes, um nó pode ter vários *filhos* (descendentes diretos), mas apenas um *pai* (ascendente direto).
 - Em relação ao nó **c**:
 - filhos: {**e**, **f**} - e *irmãos* entre si, e;
 - descendentes: {**e**, **f**, **h**, **i**, **j**}.



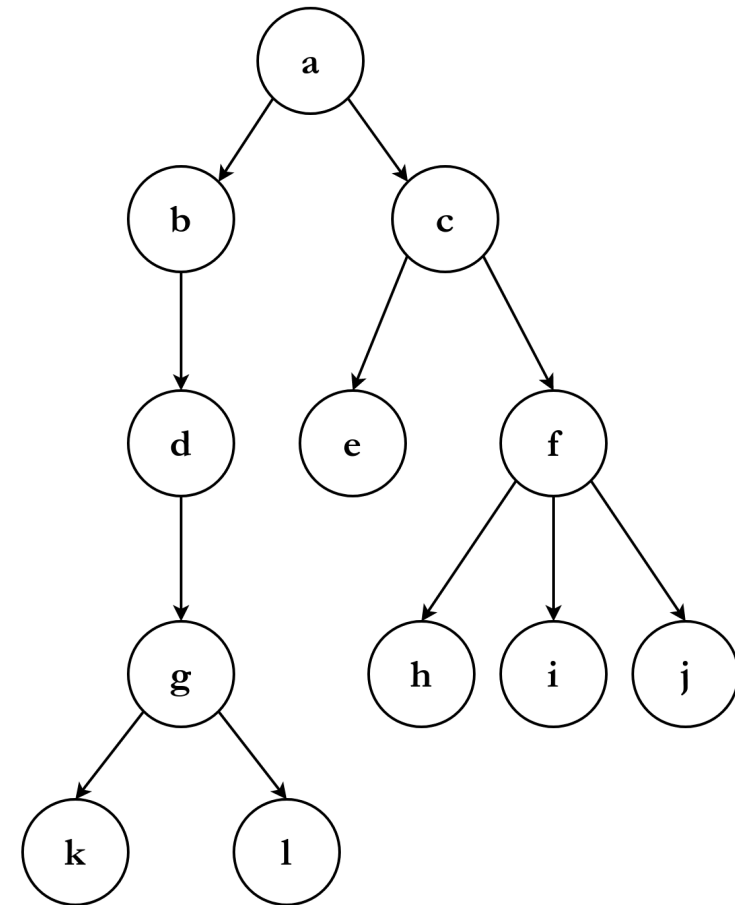
Árvores | Conceitos

- Nós que não têm descendentes são chamados de **nós externos** ou **folhas**.
 - {e, k, l, h, i, j}
- Nós que **não são a raiz** e **não são folhas** são chamados de **nós internos**.
 - {b, c, d, f, g}



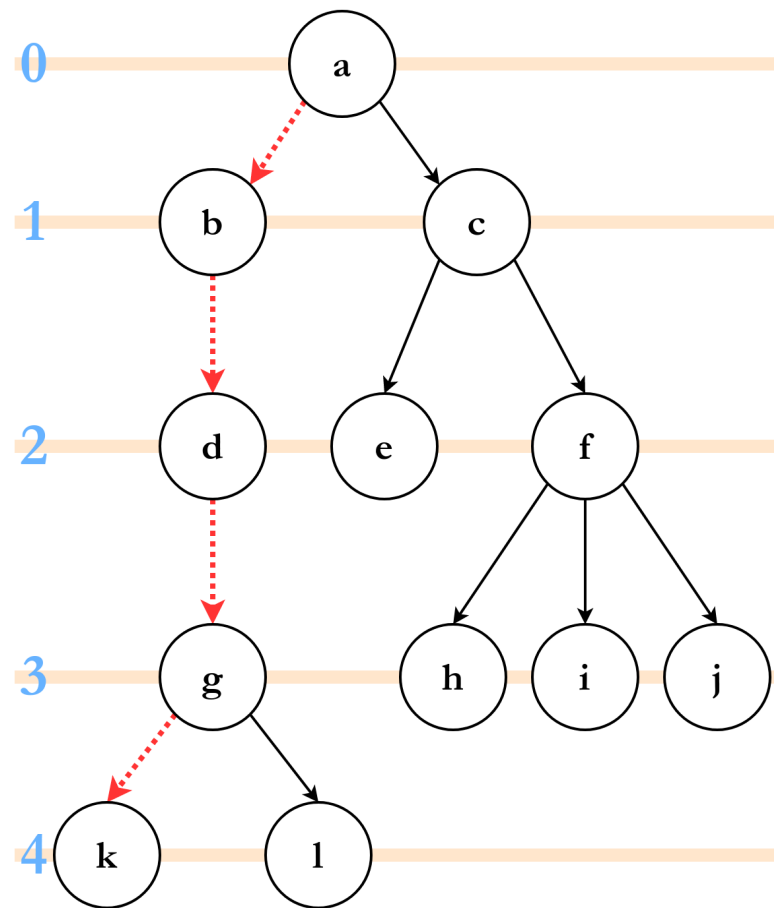
Árvores | Grau e Ordem

- O **grau de um nó** corresponde ao seu número de filhos
 - e.g., $a : 2, b : 1, f : 3$
- A **ordem de uma árvore** consiste no grau máximo permitido para os seus nós.
 - e.g., **árvores binárias** são de ordem 2.



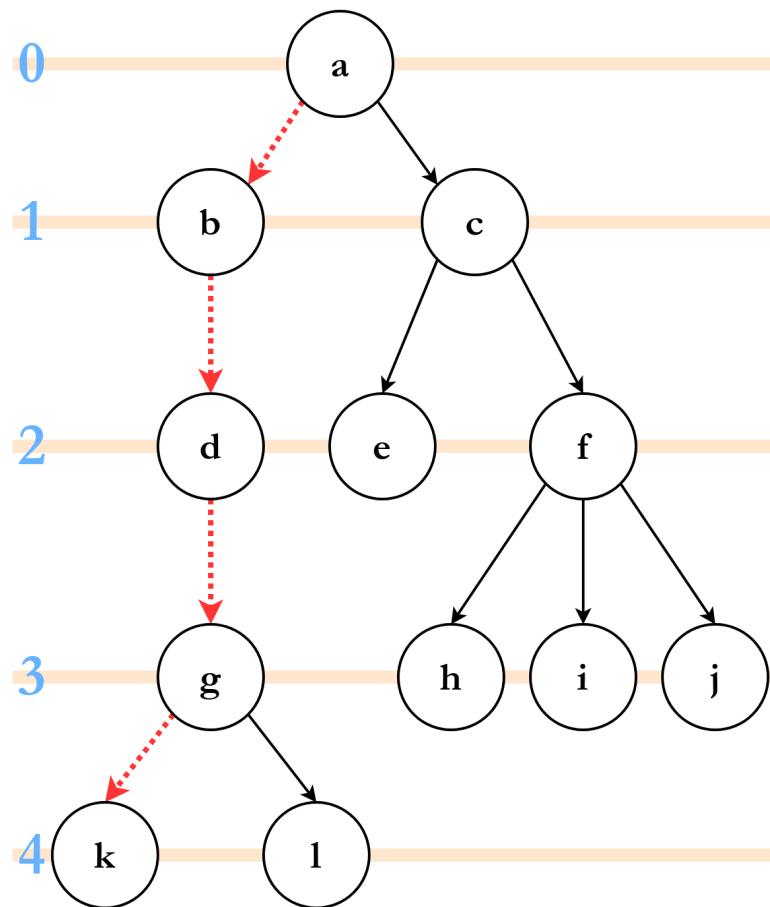
Árvores | Níveis e Altura

- As árvores podem ser organizadas em **níveis**
 - Raiz está no nível 0;
 - O nível 2 contém todos os filhos do nível 1, etc.
- A **altura da árvore** corresponde ao maior nível da árvore
 - (ou, equivalente) ao maior caminho presente na árvore;
 - Na figura ➡, a altura é 4;



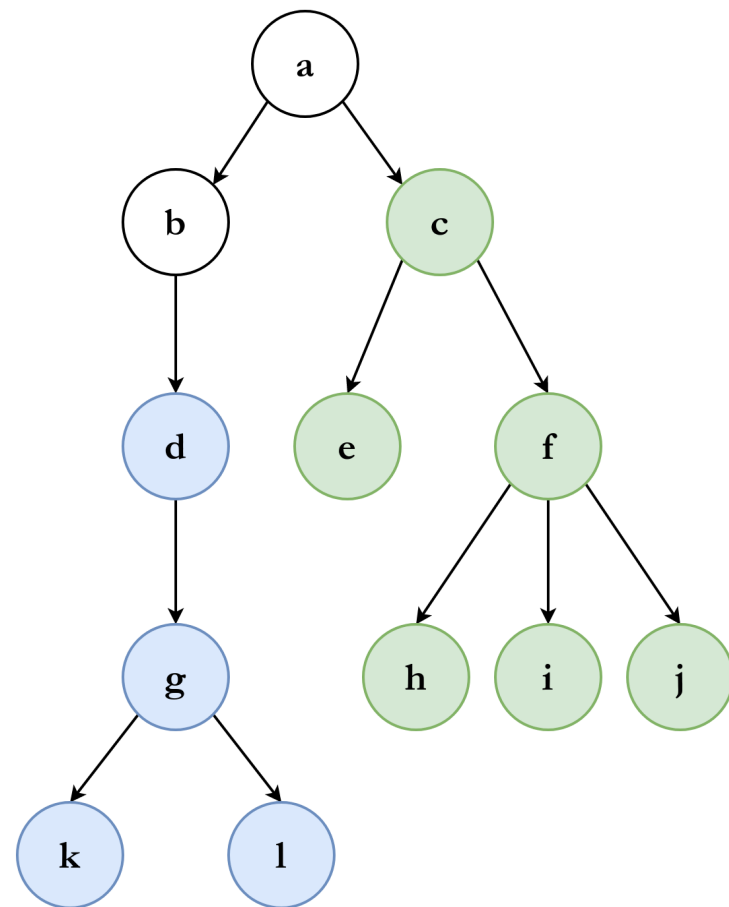
Árvores | Níveis e Altura

- A altura da árvore corresponde ao maior nível da árvore
 - (...)
 - Uma árvore "vazia" tem altura -1;
 - Uma árvore contendo apenas a raiz tem altura 0 (zero).



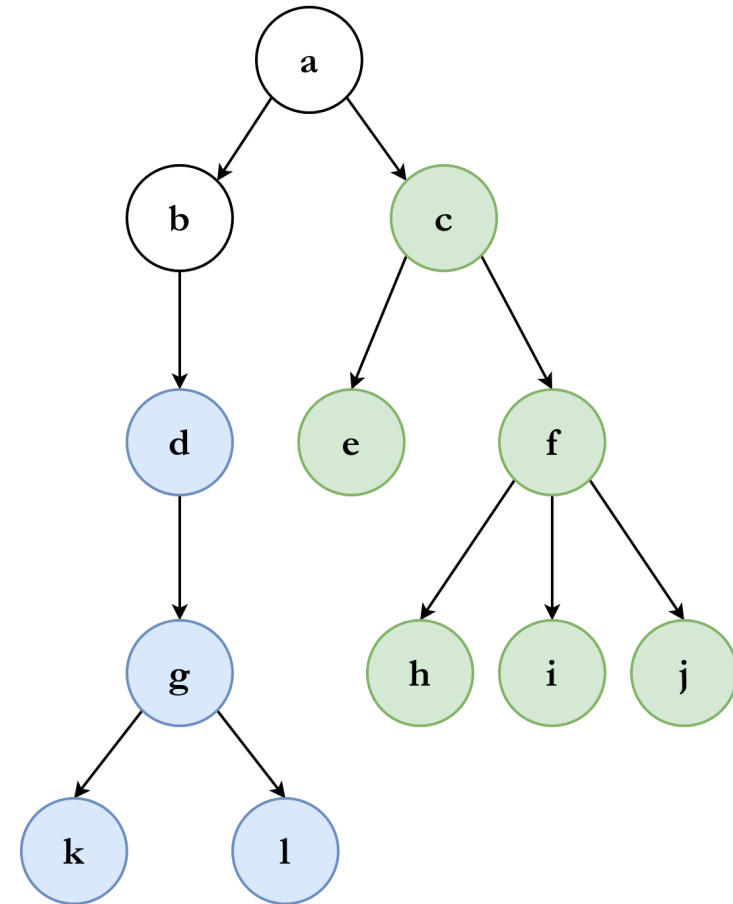
Árvores | subárvores

- Dado uma árvore e um nó n , o conjunto de todos os nós que possuem n como ascendente é chamada a **subárvore com raiz em n** .
- Exemplos:
 - subárvore com raiz em d
 - subárvore com raiz em c



Árvores | subárvores

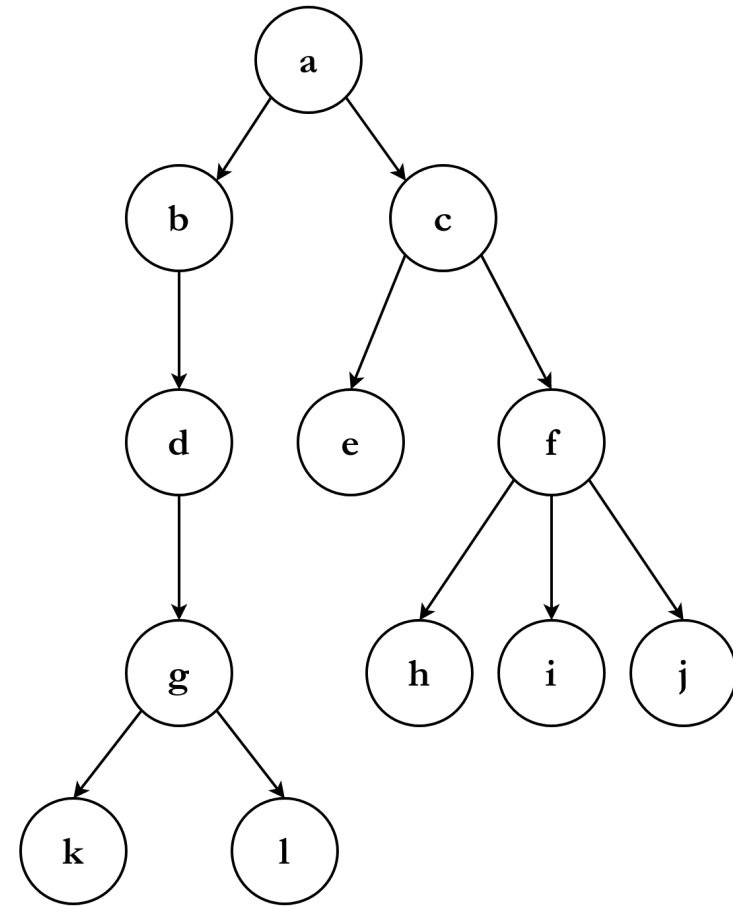
- Isto permite a abstração **recursiva** de uma árvore:
 - Uma árvore é composta por um nó (raiz) que possui um determinado número de filhos, que por sua vez representam árvores menores.



Árvores | Travessia

Em **largura** (*breadth-first traversal*):

- [esq/dir] a b c d e f g h i j k l
- [dir/esq] a c b f e d j i h g l k

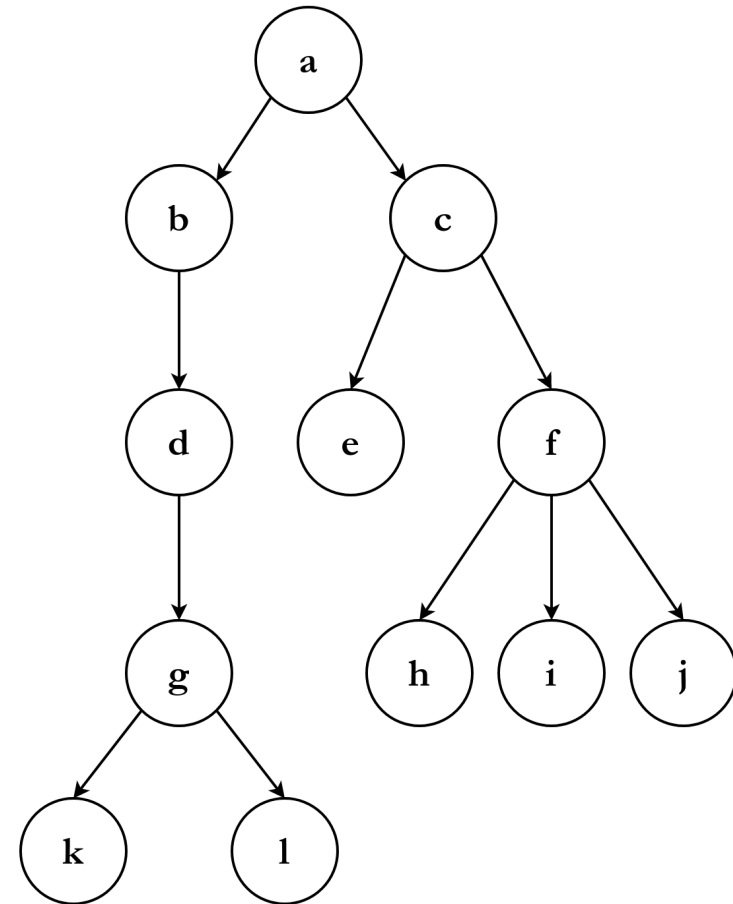


Árvores | Travessia

Em **profundidade** (*depth-first traversal*):

- [pré-ordem] **a b d g k l c e f h i j**
 - Os nós são visitados antes dos seus descendentes.
- [pós-ordem] **k l g d b e h i j f c a**
 - Os nós são visitados depois dos seus descendentes.

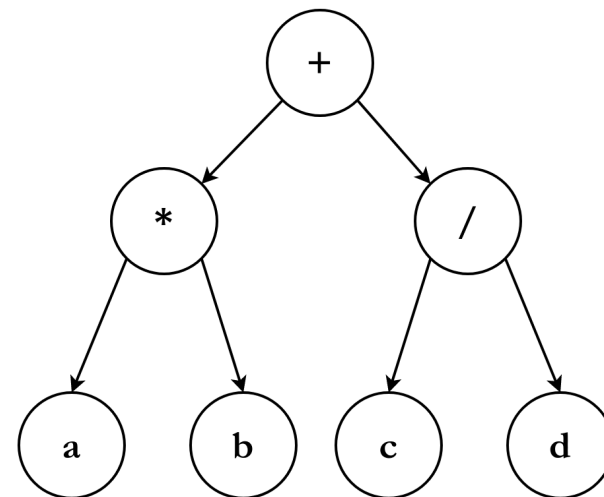
Nota: Nos exemplos os descendentes são visitados da esquerda para a direita.



Árvores | Árvores Binárias

Consistem em árvores de **ordem 2**, i.e., cada nó pode ser no máximo de *grau 2* (máx. dois filhos).

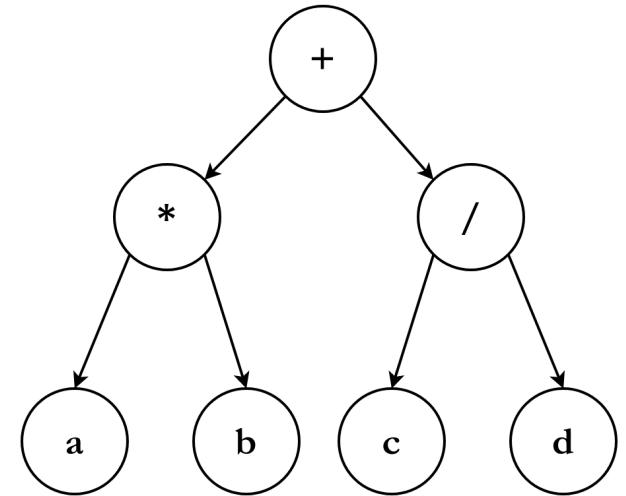
Na figura ➡ um exemplo de uma árvore binária para representar uma expressão matemática.



Árvores | Árvores Binárias

A **travessia** de árvores binárias contempla um modo adicional **em-ordem** (antes de um nó ser visitado é visitado o seu filho esquerdo; e no final o filho direito) :

- [em-ordem]: `a * b + c / d`
 - *Notação Convencional*
- [pré-ordem]: `+ * a b / c d`
 - *Notação Polaca*
- [pós-ordem]: `a b * c d / +`
 - *Notação Polaca Invertida*



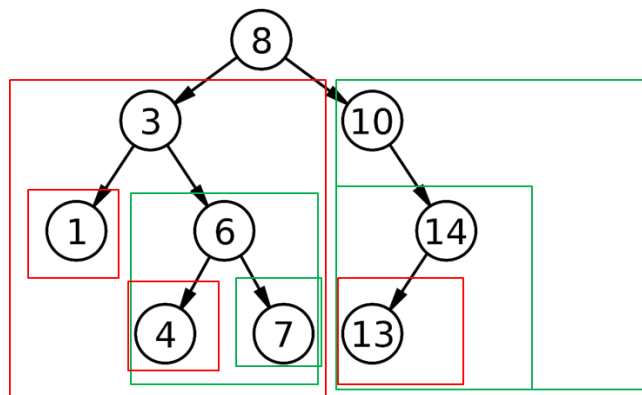
Exercícios (1)

Elabore a ficha de atividades disponível no Moodle:

`2a_FichaAtividades.pdf`

Árvores e algoritmos recursivos

- É a característica **recursiva** intrínseca à composição de uma árvore a base para a definição de soluções recursivas nos algoritmos que manipulam estruturas de dados do tipo árvore.
- Uma árvore é composta por **zero nós (vazia)** ou por **um nó (raiz)** que possui **um determinado número de filhos**, que por sua vez representam árvores menores (subárvores).



Algoritmo recursivo (1)

Algoritmo recursivo para percorrer uma árvore binária

Algoritmo pre-order numa árvore binária : [Raiz, subárvore esquerda, subárvore direita]

```
Algorithm: preOrder
    input: binaryTree
    output :None

BEGIN
    IF NOT(isEmpty(binaryTree) THEN
        WRITE(root(binaryTree))
        preOrder(leftTree(binaryTree))
        preOrder(rightTree(binaryTree))
    END IF
END
```

Algoritmo recursivo (2)

Algoritmo recursivo para percorrer uma árvore genérica

Algoritmo pre-order : [Raiz, Sub-arvores]

```
Algorithm: preOrder
    input: tree
    output :None

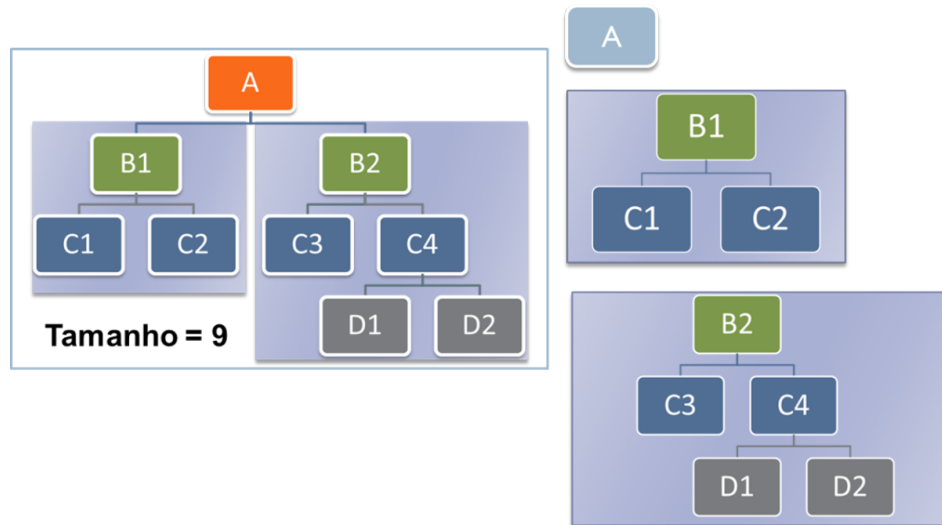
BEGIN
    IF NOT(isEmpty(tree) THEN
        WRITE(root(tree))
        FOR EACH child FROM children(tree)
            preOrder(child)
        END FOR
    END IF
END
```

Algoritmo recursivo (3)

Algoritmo para o cálculo do número de elementos de uma árvore binária

O tamanho de uma árvore é igual a:

- 0 se a árvore está vazia
- 1 + o tamanho da subárvore direita + o tamanho da sub árvore esquerda, caso contrário



Algoritmo recursivo (3) (cont)

```
Algorithm: size
  input: binaryTree
  output :natural - number of elements

BEGIN
  IF (isEmpty(binaryTree)) THEN
    RETURN 0
  ELSE
    RETURN 1 + size(leftTree(binaryTree))
    + size(rightTree(binaryTree))
  END IF
END
```

Exercícios (2)

Elabore em pseudocodigo os seguintes algoritmos:

- `exist (binaryTree, elem)` que recebe um elemento e uma árvore binária e devolve true se o elemento existe na árvore
- `height(binaryTree)` - que recebe como argumento uma árvore binária e devolve a sua altura.

Dica - a altura de uma árvore é igual ao máximo entre a altura da subárvore esquerda, e a altura da subárvore direita.

Exercícios (3) (Extra aula)

- Modifique os algoritmos recursivos, size, exist e height de forma a receber uma árvore generica em vez de uma árvore binária.

Relembrar - Uma árvore genérica é composta por uma raiz e por zero ou mais subárvores.

- Exemplo: O tamanho de uma árvore é igual a zero se a árvore for vazia , caso contrário o tamanho é igual a $(1 + \text{o somatório do tamanho de cada uma das suas subárvores})$.