

TECNOLOGIA EM INFORMÁTICA

Estrutura de Dados

Prof. Glauco da Silva
contato: glaucoslv@gmail.com

Roteiro

- *Objetivos*
- *Árvores*
- *Árvores binárias*
- *Atravessamento em árvores binárias*
- *Exercícios*

Roteiro

- ***Objetivos***
- *Árvores*
- *Árvores binárias*
- *Atravessamento em árvores binárias*
- *Exercícios*

Objetivos

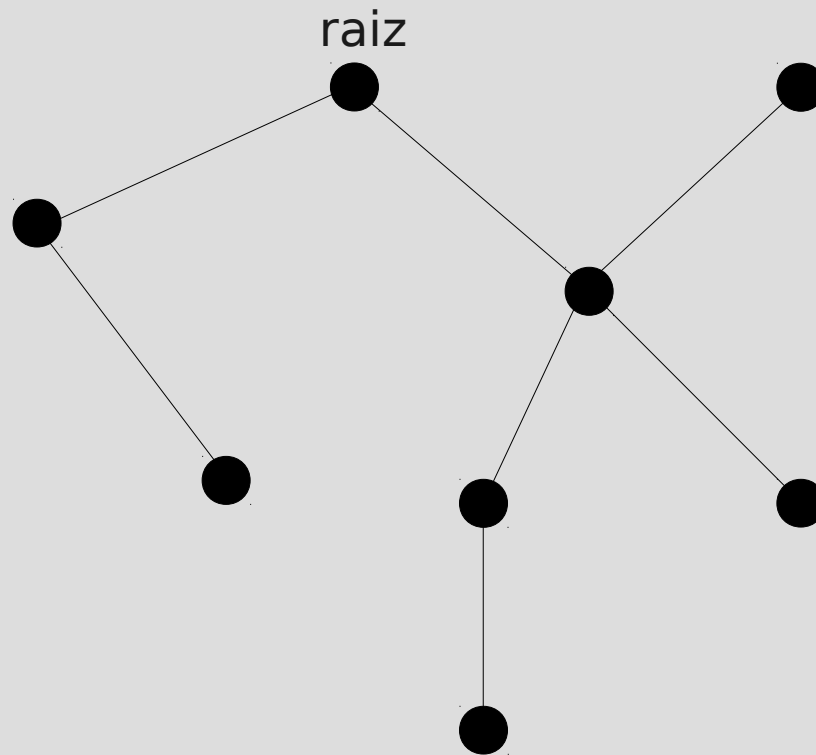
- *Apresentar os conceitos de árvores e árvores binárias;*
- *Apresentar métodos de atravessamento em árvores binárias.*

Roteiro

- *Objetivos*
- *Árvores*
- *Árvores binárias*
- *Atravessamento em árvores binárias*
- *Exercícios*

Árvores

- *Estruturas utilizadas quando se trabalha com hierarquias*



Árvores

- *Definição*

- *Uma árvore T é um conjunto finito de nós (ou vértices) tal que:*

- *$T = \emptyset$, isto é, uma árvore vazia*

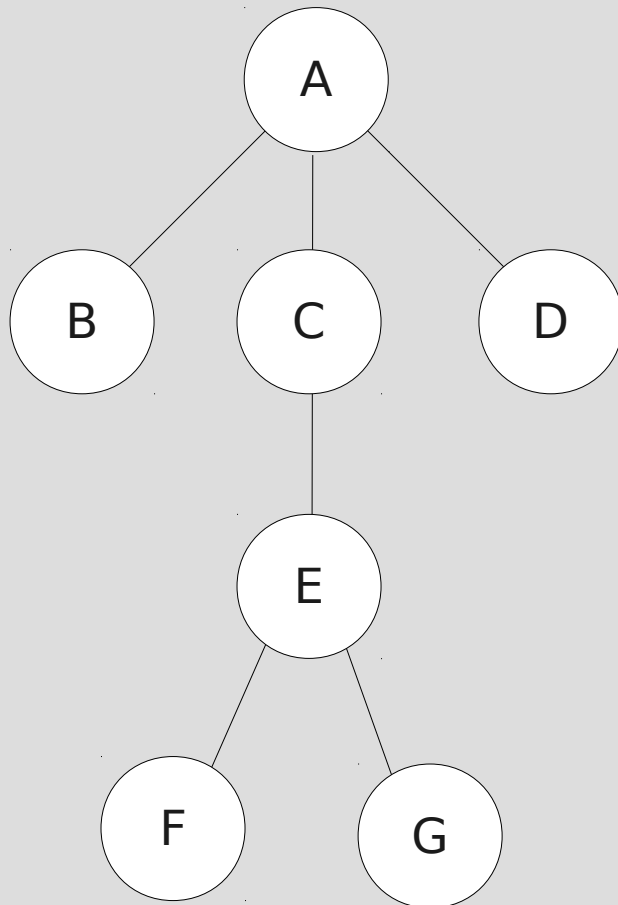
- *Um nó raiz e um conjunto de árvores não vazias, chamadas de subárvores do nó raiz*

- *É comum associar rótulos aos nós das árvores*

- *Na prática, os nós são usados para armazenar informações diversas*

Árvores

- *Representação*



Árvores

- *Nomenclatura*

- *Raiz*

- *O nó na parte superior da árvore. Cada árvore possui somente um nó raiz*

- *Pai*

- *Qualquer nó, com exceção da raiz possui um nó pai, é o nó imediatamente antes dele*

- *Filho*

- *Qualquer nó abaixo de um dado nó*

- *Folha*

- *Um nó que não tem filhos*

- *Subárvore*

- *Árvore formada a partir de um filho de qualquer nó*

Árvores

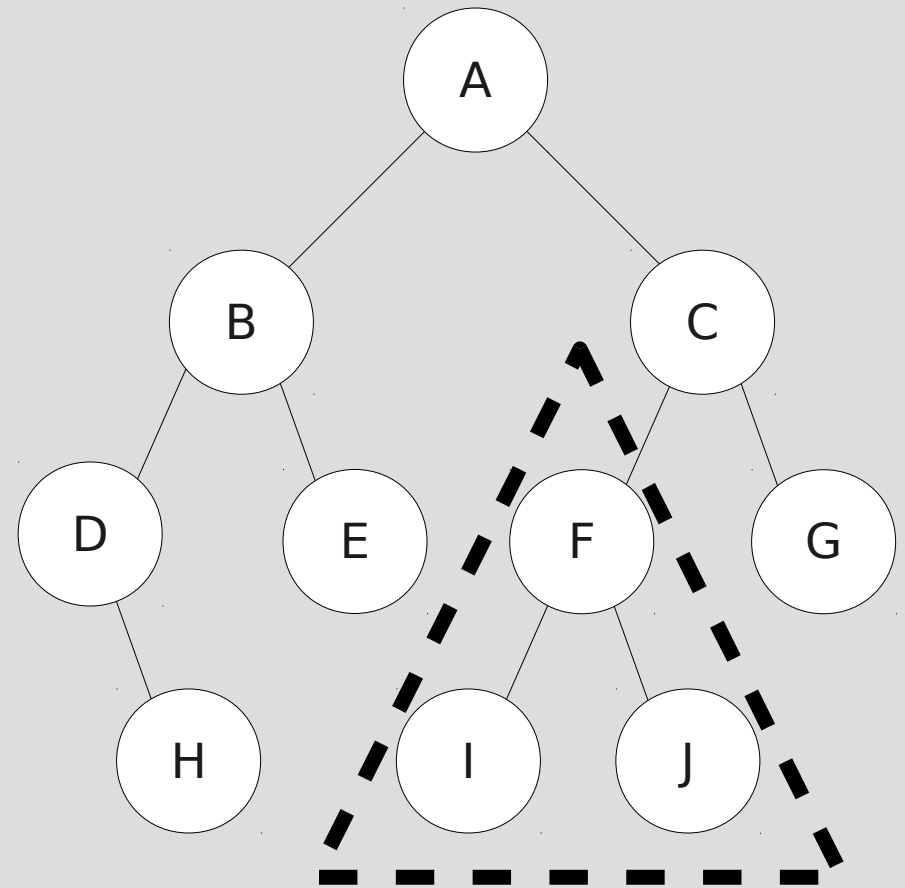
- *Nomenclatura*

A: Raíz da árvore

B e C: Filhos de A

H, E, G, I, J: Folhas

F e G: Filhos de C



Subárvore tendo F
como raiz

Árvores

- *Nomenclatura*

- *Caminho*

- *Sequência resultante da passagem por nós*

- *Nível*

- *Posição onde o nó se encontra na árvore. A raiz está no nível 0*

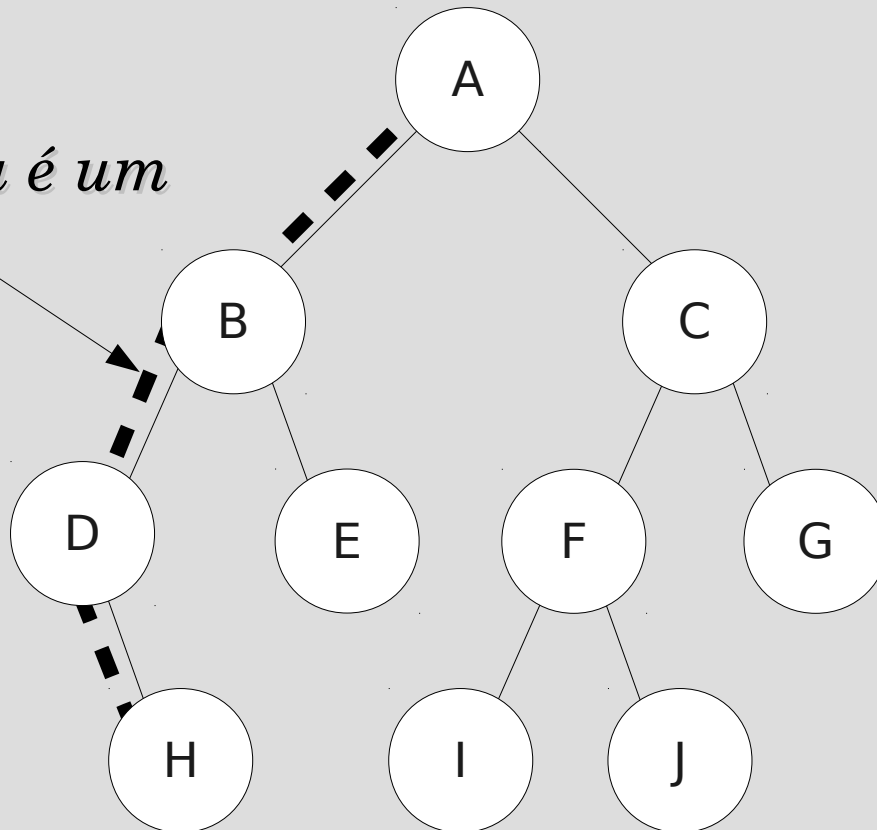
- *Altura*

- *O maior nível da árvore indica sua altura*

Árvores

- *Nomenclatura*

A linha tracejada é um caminho



← *Nível 0*

← *Nível 1*

← *Nível 2*

← *Nível 3*

Altura da árvore: 3

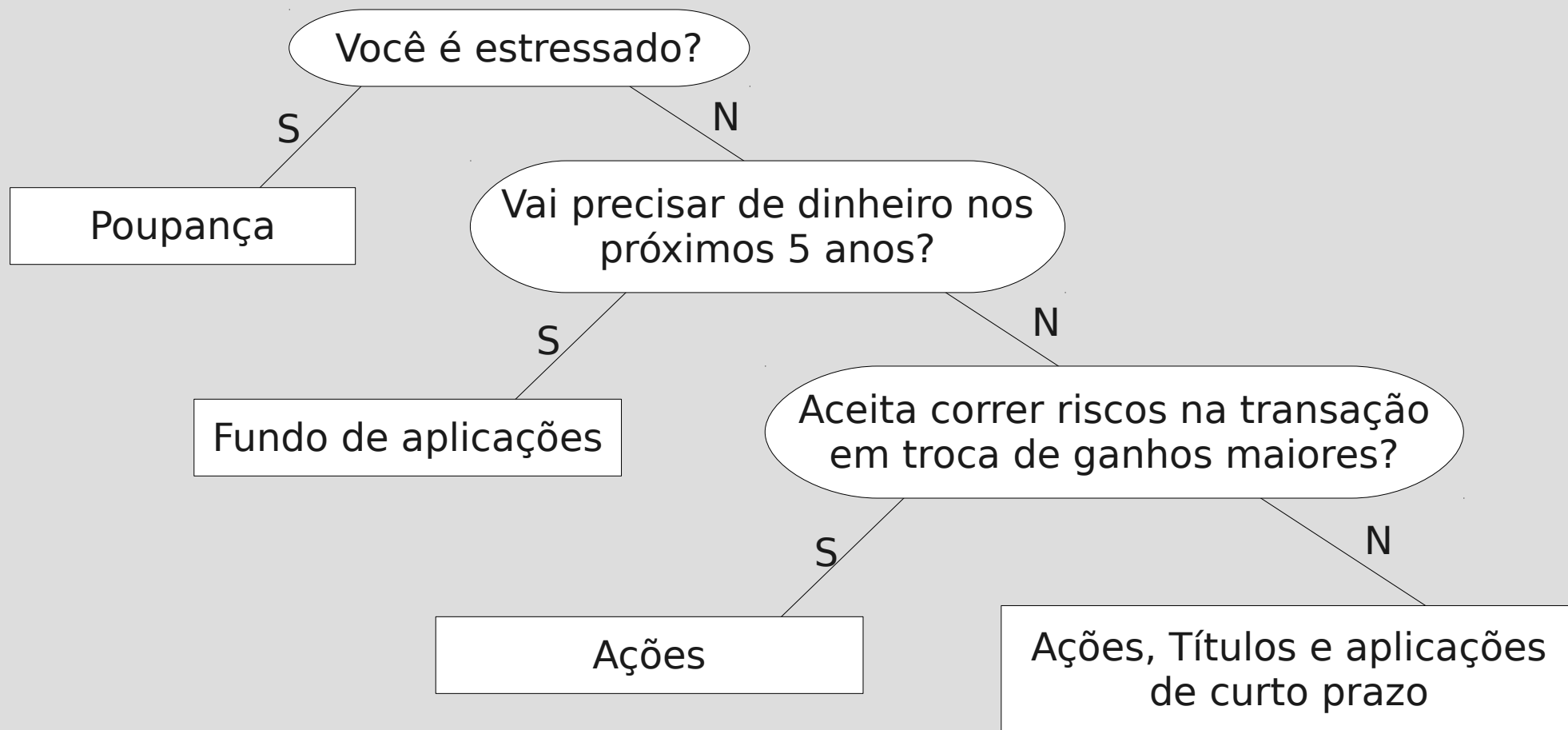
Roteiro

- *Objetivos*
- *Árvores*
- ***Árvores binárias***
- *Atravessamento em árvores binárias*
- *Exercícios*

Árvores Binárias

- *Árvore onde todo nó pode ter no máximo 2 filhos*
- *Os filhos são chamados de filho à esquerda e filho à direita*
 - *Estrutura útil quando uma de duas decisões deve ser tomada no decorrer do processo.*

Árvores Binárias



Árvores Binárias

- *Exemplo: Encontrar números duplicados em uma lista dada*
 - *Armazenam-se os números na árvore de forma a:*
 - *o 1º número é armazenado na raiz de uma árvore binária*
 - *cada um dos próximos números na lista é comparado com a raiz:*
 - caso seja igual é uma duplicata*
 - caso seja menor, é armazenado na sub-árvore da esquerda seguindo-se recursivamente o mesmo procedimento*
 - caso seja maior, é armazenado na sub-árvore da direita seguindo-se recursivamente o mesmo procedimento*

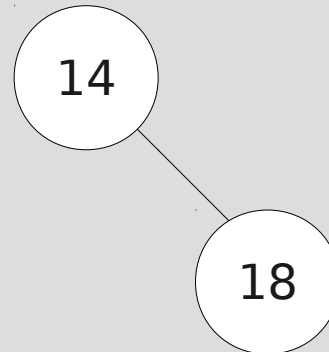
Árvores Binárias

- *Exemplo: Armazenar a seguinte lista em uma árvore binária: 14, 18, 4, 9, 7, 15, 3, 4, 20, 9, 5*

14

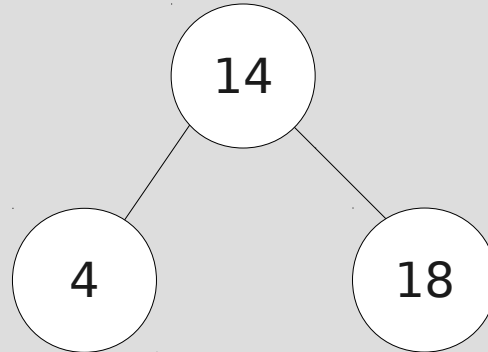
Árvores Binárias

- *Exemplo: Armazenar a seguinte lista em uma árvore binária: 14, 18, 4, 9, 7, 15, 3, 4, 20, 9, 5*



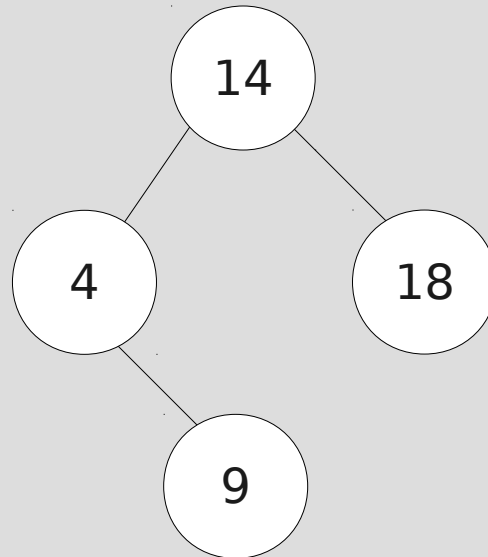
Árvores Binárias

- *Exemplo: Armazenar a seguinte lista em uma árvore binária: 14, 18, 4, 9, 7, 15, 3, 4, 20, 9, 5*



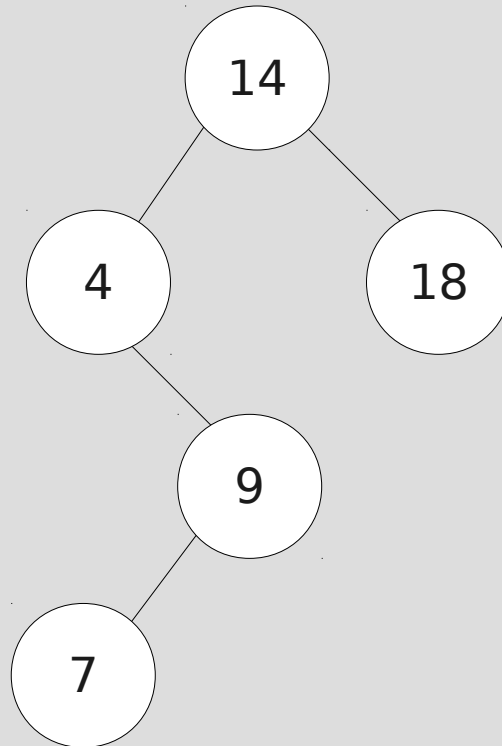
Árvores Binárias

- *Exemplo: Armazenar a seguinte lista em uma árvore binária: 14, 18, 4, 9, 7, 15, 3, 4, 20, 9, 5*



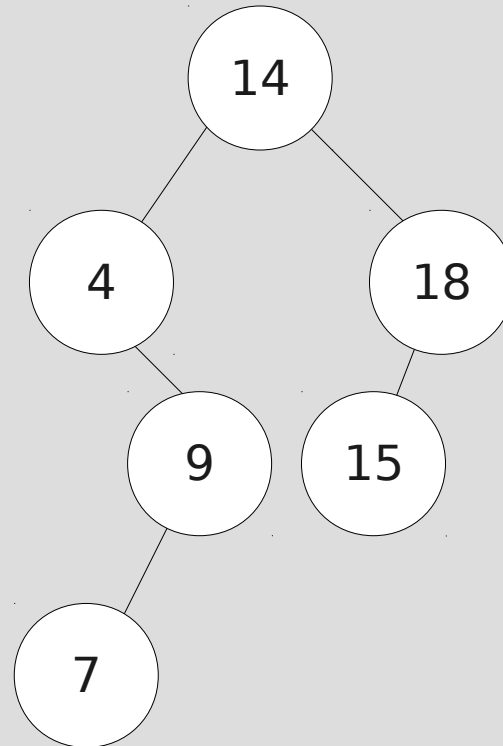
Árvores Binárias

- *Exemplo: Armazenar a seguinte lista em uma árvore binária: 14, 18, 4, 9, 7, 15, 3, 4, 20, 9, 5*



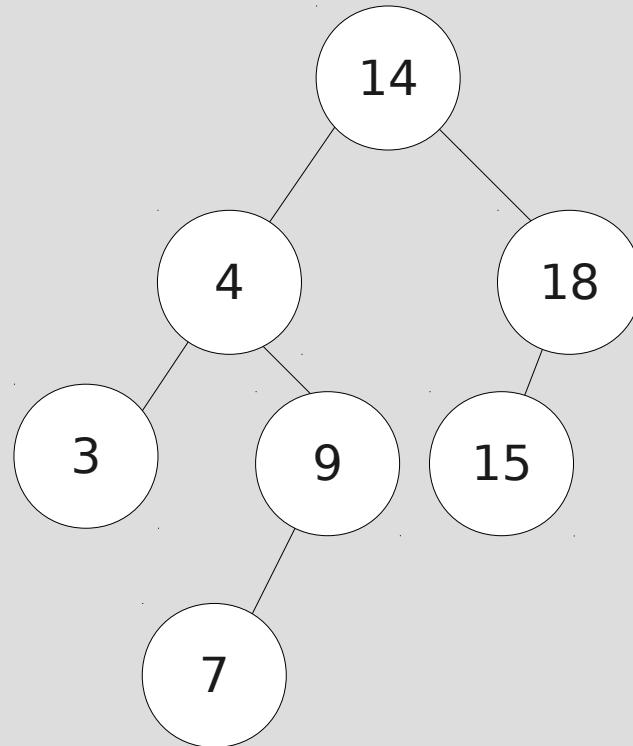
Árvores Binárias

- *Exemplo: Armazenar a seguinte lista em uma árvore binária: 14, 18, 4, 9, 7, 15, 3, 4, 20, 9, 5*



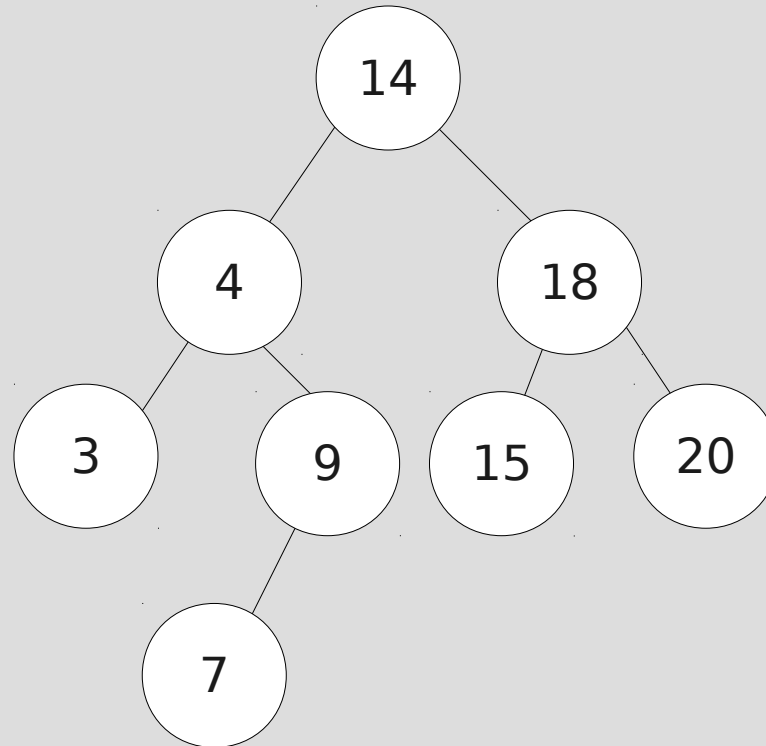
Árvores Binárias

- *Exemplo: Armazenar a seguinte lista em uma árvore binária: 14, 18, 4, 9, 7, 15, 3, 4, 20, 9, 5*



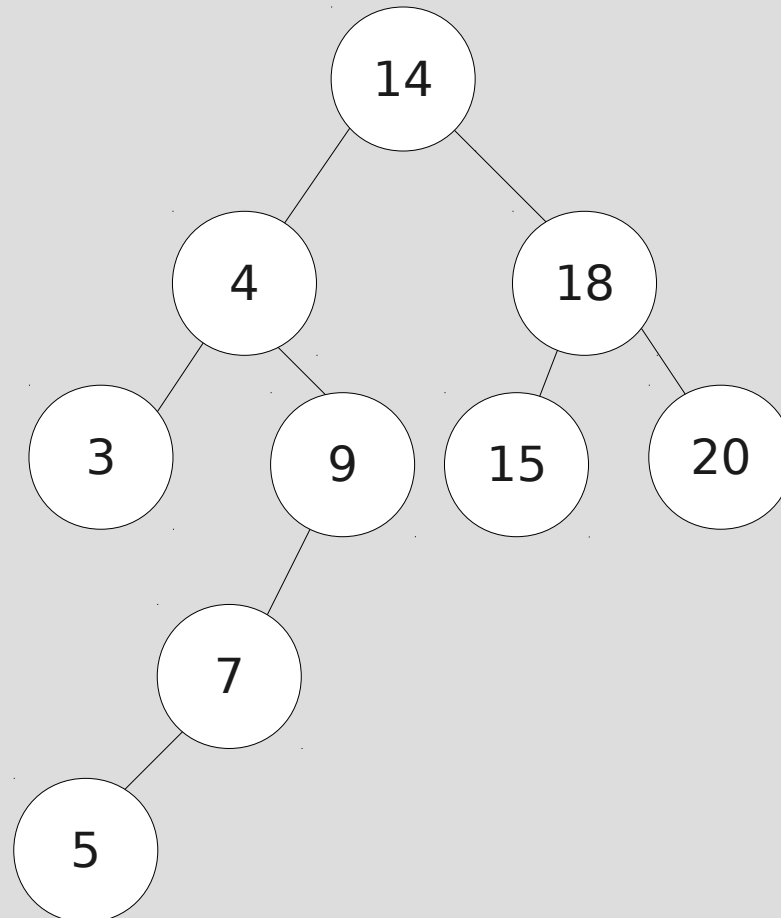
Árvores Binárias

- *Exemplo: Armazenar a seguinte lista em uma árvore binária: 14, 18, 4, 9, 7, 15, 3, 4, 20, 9, 5*



Árvores Binárias

- *Exemplo: Armazenar a seguinte lista em uma árvore binária: 14, 18, 4, 9, 7, 15, 3, 4, 20, 9, 5*



Roteiro

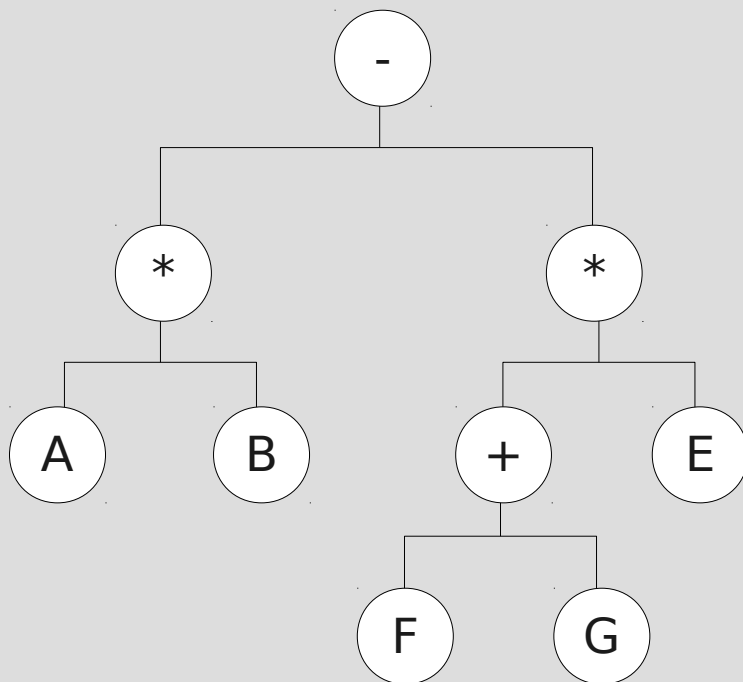
- *Objetivos*
- *Árvores*
- *Árvores binárias*
- ***Atravessamento em árvores binárias***
- *Exercícios*

Atravessamento em Árvores Binárias

- *O atravessamento de uma árvore consiste em visitar todos os nós da árvore*
- *Existem três formas de se fazer atravessamentos*
 - *Pré-ordem: visite a raiz, em seguida visite a subárvore da esquerda, depois a subárvore da direita*
 - *Em-ordem: visite a subárvore da esquerda, a raiz e a subárvore da direita*
 - *Pós-ordem: visite a subárvore da esquerda, a subárvore da direita e então a raiz*

Atravessamento em Árvores Binárias

- *Exemplo: Atravessar a árvore que representa a expressão $A * B - (F + G) * E$*



pré-ordem:

$- * A B * + F G E$

em-ordem:

$A * B - F + G * E$

pós-ordem:

$A B * F G + E * -$

Atravessamento em Árvores Binárias

- *Pré-ordem*

```
pre_ordem (pt)
{
    if (pt == NULL) return ();
    visite(raiz);
    pre_ordem (pt->esq);
    pre_ordem (pt-> dir);
}
```

Atravessamento em Árvores Binárias

- *Em-ordem*

```
em_ordem (pt)
{
    if (pt == NULL) return ();
    em_ordem (pt->esq);
    visite(raiz);
    em_ordem (pt-> dir);
}
```

Atravessamento em Árvores Binárias

- *Pós-ordem*

```
pos_ordem (pt)
{
    if (pt == NULL) return ();
    pos_ordem (pt->esq);
    pos_ordem (pt-> dir);
    visite(raiz);
}
```

Roteiro

- *Objetivos*
- *Árvores*
- *Árvores binárias*
- *Atravessamento em árvores binárias*
- ***Exercícios***

Exercícios

- *A partir de uma árvore binária vazia, mostre o estado final após a inserção dos seguintes itens nessa ordem: 23, 14, 8, 4, 30, 34, 31, 22, 10, 70, 45, 67. Em seguida determine o nível de cada elemento e a altura da árvore.*

Exercícios

- *Faça os atravessamentos Pré-ordem, Em-ordem e Pós-ordem na seguinte árvore binária*

