

# Árvores

Algoritmos, Estruturas de Dados,  
Programação Imperativa e C

Sérgio Soares  
scbs@cin.ufpe.br

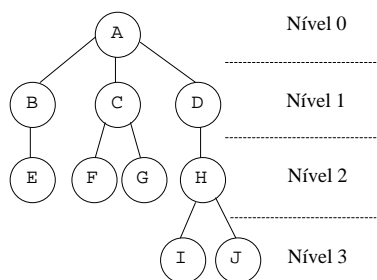
# Árvores

- Armazena elementos em uma relação de hierarquia.
  - conjunto de dados subordinados a outro
- Compostas de nós responsáveis por armazenar os dados
  - nó raiz
  - Subárvores a partir do nó raiz e dos subnós subsequentes

Slide 2

©Sérgio Soares, todos os direitos reservados

# Árvores – nomenclatura



Nó	Grau
A	3
B	1
C	2
D	1
E	0
F	0
G	0
H	2
I	0

Slide 3

©Sérgio Soares, todos os direitos reservados

# Árvores – nomenclatura

- Altura
  - nível mais alto da árvore
- Árvore ordenada
  - ordem das subárvores é significativa
- Árvore orientada
  - ordem das subárvores não é significativa
- Pai
  - nó com duas subárvores
- Irmãos
  - nós com o mesmo pai

Slide 4

©Sérgio Soares, todos os direitos reservados

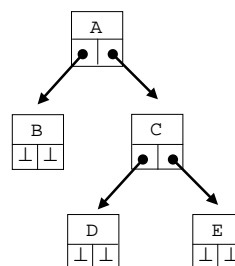
# Árvores

- Binárias
  - cada nó tem no máximo dois filhos (grau menor ou igual a 2)
  - filhos se distinguem entre direita e esquerda

Slide 5

©Sérgio Soares, todos os direitos reservados

# Árvores binárias – representação



Slide 6

©Sérgio Soares, todos os direitos reservados

## Percorrendo árvores

- Pré-fixado – RED
- Central – ERD
- Pós-fixado – EDR

E – percorra subárvore da esquerda

R – visita a raiz

D – percorra subárvore da direita

Utilizar recursividade

Slide 7

©Sérgio Soares, todos os direitos reservados

## Algoritmo de varredura

```
proc imprimePreFixado(Arvore a)
  se a ≠ 1 então
    escreva(a.dado);
    imprimePreFixado(a.esquerda);
    imprimePreFixado(a.direita);
```

Slide 8

©Sérgio Soares, todos os direitos reservados

## Árvores Binárias de busca

- Armazena elementos que tem uma relação de ordem (>, <, =) entre si
- Se a árvore for vazia, o elemento fica na raiz.
- Se não, se o elemento for menor fica na subárvore da esquerda, e se for maior na direita.

Slide 9

©Sérgio Soares, todos os direitos reservados

## Exercício

- 1 – Escreva um procedimento que dada uma árvore binária de pesquisa e um número inteiro, insere o número na árvore. Quantos nós são percorridos, no máximo, para inserir o elemento.

```
proc inserir(Arvore a, int num)
```

Slide 10

©Sérgio Soares, todos os direitos reservados

## Exercício

- 2 - Escreva um programa que lê uma seqüência de 10 números e os armazena em uma árvore binária de busca, utilizando o procedimento definido no exercício anterior. Em seguida o programa imprime os números de maneira pós-fixada.

Slide 11

©Sérgio Soares, todos os direitos reservados

## Exercício

- 3 - Escreva um procedimento que verifica se um dado número está armazenado numa dada árvore.

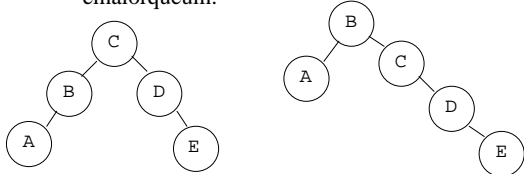
```
proc bool existe(Arvore a, int num)
```

Slide 12

©Sérgio Soares, todos os direitos reservados

## Árvores balanceadas

- Árvore organizada de forma que a distância média dos nós até a raiz é mínima.
  - a diferença de altura entre suas subárvores não é maior que um.



Slide 13

©Sérgio Soares, todos os direitos são reservados