

Estruturas de Dados

Cristina Gomes Fernandes

Árvores

Árvore enraizada T : conjunto finito de elementos denominados **nós** ou vértices, tais que:

- $T = \emptyset$
- existe um nó especial r em T , chamado de **raiz**, e uma partição dos demais nós de T em m conjuntos não-vazios, cada um deles sendo uma árvore.

Árvores

Árvore enraizada T : conjunto finito de elementos denominados **nós** ou vértices, tais que:

- $T = \emptyset$
- existe um nó especial r em T , chamado de **raiz**, e uma partição dos demais nós de T em m conjuntos não-vazios, cada um deles sendo uma árvore.

Cada um dos m conjuntos é chamado de **subárvore** de T .

Árvores

Árvore enraizada T : conjunto finito de elementos denominados **nós** ou vértices, tais que:

- $T = \emptyset$
- existe um nó especial r em T , chamado de **raiz**, e uma partição dos demais nós de T em m conjuntos não-vazios, cada um deles sendo uma árvore.

Cada um dos m conjuntos é chamado de **subárvore** de T .

A raiz de cada uma destas subárvores é um **filho** de r e r é o **pai** destas raízes.

O número de filhos de um nó é seu **grau**.

Um nó de grau zero é uma **folha**.

Árvores

Árvore enraizada T : conjunto finito de elementos denominados **nós** ou vértices, tais que:

- $T = \emptyset$
- existe um nó especial r em T , chamado de **raiz**, e uma partição dos demais nós de T em m conjuntos não-vazios, cada um deles sendo uma árvore.

Cada um dos m conjuntos é chamado de **subárvore** de T .

A raiz de cada uma destas subárvores é um **filho** de r e r é o **pai** destas raízes.

O número de filhos de um nó é seu **grau**.

Um nó de grau zero é uma **folha**.

A árvore é **ordenada** se há ordem nos filhos de cada nó.

Árvores

Os nós de uma subárvore de raiz x são os **descendentes** de x , e x é um **ascendente** de cada um destes nós.

Árvores

Os nós de uma subárvore de raiz x são os **descendentes** de x , e x é um **ascendente** de cada um destes nós.

O **nível** de um nó é definido recursivamente:

- o nível da raiz é 0,
- o nível de um nó que não é a raiz é 1 a mais que o nível de seu pai.

Árvores

Os nós de uma subárvore de raiz x são os **descendentes** de x , e x é um **ascendente** de cada um destes nós.

O **nível** de um nó é definido recursivamente:

- o nível da raiz é 0,
- o nível de um nó que não é a raiz é 1 a mais que o nível de seu pai.

A **altura** de uma árvore é o nível de um nó de nível máximo na árvore.

A altura de um nó x é a altura da subárvore de raiz x .

Árvores

Os nós de uma subárvore de raiz x são os **descendentes** de x , e x é um **ascendente** de cada um destes nós.

O **nível** de um nó é definido recursivamente:

- o nível da raiz é 0,
- o nível de um nó que não é a raiz é 1 a mais que o nível de seu pai.

A **altura** de uma árvore é o nível de um nó de nível máximo na árvore.

A altura de um nó x é a altura da subárvore de raiz x .

Se o grau de todo nó é no máximo 2, a árvore é **binária**.

Os filhos em uma árvore binária são chamados **esquerdo** e **direito**.

Árvores

Uma árvore binária pode ser representada com apontadores, usando uma célula com o seguinte formato:

- **info**: valor armazenado
- **pai**: apontador para o pai deste nó
- **esq**: apontador para o filho esquerdo
- **dir**: apontador para o filho direito.

Algumas rotinas

CONTENÓS (T)

1 **se** $T = \text{NIL}$

2 **então devolva** 0

3 **senão devolva**

$1 + \text{CONTENÓS}(\text{esq}(T)) + \text{CONTENÓS}(\text{dir}(T))$

Algumas rotinas

CONTENÓS (T)

- 1 **se** $T = \text{NIL}$
- 2 **então devolva** 0
- 3 **senão devolva**
 $1 + \text{CONTENÓS}(\text{esq}(T)) + \text{CONTENÓS}(\text{dir}(T))$

IMPRIMENÍVEL (T, i)

- 1 **se** $T \neq \text{NIL}$
- 2 **então se** $i = 0$
- 3 **então imprima** $\text{info}(T)$
- 4 **senão** $\text{IMPRIMENÍVEL}(\text{esq}(T), i - 1)$
- 5 $\text{IMPRIMENÍVEL}(\text{dir}(T), i - 1)$

Algumas rotinas

CONTENÓS (T)

- 1 **se** $T = \text{NIL}$
- 2 **então devolva** 0
- 3 **senão devolva**
 $1 + \text{CONTENÓS}(\text{esq}(T)) + \text{CONTENÓS}(\text{dir}(T))$

IMPRIMENÍVEL (T, i)

- 1 **se** $T \neq \text{NIL}$
- 2 **então se** $i = 0$
- 3 **então imprima** $\text{info}(T)$
- 4 **senão** $\text{IMPRIMENÍVEL}(\text{esq}(T), i - 1)$
- 5 $\text{IMPRIMENÍVEL}(\text{dir}(T), i - 1)$

Quanto tempo consomem?

Mais rotinas

CONTEFOLHAS (T)

1 **se** $T = \text{NIL}$

2 **então devolva** 0

3 **senão se** $esq(T) = \text{NIL}$ **e** $dir(T) = \text{NIL}$

4 **então devolva** 1

5 **senão devolva**

CONTEFOLHAS($esq(T)$) + **CONTEFOLHAS**($dir(T)$)

Mais rotinas

CONTEFOLHAS (T)

1 **se** $T = \text{NIL}$

2 **então devolva** 0

3 **senão se** $esq(T) = \text{NIL}$ **e** $dir(T) = \text{NIL}$

4 **então devolva** 1

5 **senão devolva**

$\text{CONTEFOLHAS}(esq(T)) + \text{CONTEFOLHAS}(dir(T))$

ALTURA (T, i)

1 **se** $T = \text{NIL}$

2 **então devolva** -1

3 **senão** $h_e \leftarrow \text{ALTURA}(esq(T))$

4 $h_d \leftarrow \text{ALTURA}(dir(T))$

5 **devolva** $1 + \max\{h_e, h_d\}$

Mais rotinas

CONTEFOLHAS (T)

1 **se** $T = \text{NIL}$

2 **então devolva** 0

3 **senão se** $esq(T) = \text{NIL}$ **e** $dir(T) = \text{NIL}$

4 **então devolva** 1

5 **senão devolva**

$\text{CONTEFOLHAS}(esq(T)) + \text{CONTEFOLHAS}(dir(T))$

ALTURA (T, i)

1 **se** $T = \text{NIL}$

2 **então devolva** -1

3 **senão** $h_e \leftarrow \text{ALTURA}(esq(T))$

4 $h_d \leftarrow \text{ALTURA}(dir(T))$

5 **devolva** $1 + \max\{h_e, h_d\}$

Quanto tempo consomem?

Percursos em árvores binárias

PRÉORDEM (T)

1 **se** $T \neq \text{NIL}$

2 **então imprima** $\text{info}(T)$

3 PRÉORDEM ($\text{esq}(T)$)

4 PRÉORDEM ($\text{dir}(T)$)

Percursos em árvores binárias

PRÉORDEM (T)

- 1 **se** $T \neq \text{NIL}$
- 2 **então imprima** $\text{info}(T)$
- 3 PRÉORDEM ($\text{esq}(T)$)
- 4 PRÉORDEM ($\text{dir}(T)$)

PÓSORDEM (T)

- 1 **se** $T \neq \text{NIL}$
- 2 **então** PÓSORDEM ($\text{esq}(T)$)
- 3 PÓSORDEM ($\text{dir}(T)$)
- 4 **imprima** $\text{info}(T)$

Percursos em árvores binárias

PRÉORDEM (T)

- 1 **se** $T \neq \text{NIL}$
- 2 **então imprima** $\text{info}(T)$
- 3 PRÉORDEM ($\text{esq}(T)$)
- 4 PRÉORDEM ($\text{dir}(T)$)

PÓSORDEM (T)

- 1 **se** $T \neq \text{NIL}$
- 2 **então** PÓSORDEM ($\text{esq}(T)$)
- 3 PÓSORDEM ($\text{dir}(T)$)
- 4 **imprima** $\text{info}(T)$

INORDEM (T)

- 1 **se** $T \neq \text{NIL}$
- 2 **então** INORDEM ($\text{esq}(T)$)
- 3 **imprima** $\text{info}(T)$
- 4 INORDEM ($\text{dir}(T)$)

Árvores de busca binária (ABB)

Dicionário: tipo abstracto de dados para um conjunto com operações de **inserção**, **remoção** e **busca**.

ABBs servem para implementar dicionários e filas de prioridades.

Árvores de busca binária (ABB)

Dicionário: tipo abstracto de dados para um conjunto com operações de **inserção**, **remoção** e **busca**.

ABBs servem para implementar dicionários e filas de prioridades.

Cada operação consome tempo, no pior caso, proporcional à altura.

Árvores de busca binária (ABB)

Dicionário: tipo abstracto de dados para um conjunto com operações de **inserção**, **remoção** e **busca**.

ABBs servem para implementar dicionários e filas de prioridades.

Cada operação consome tempo, no pior caso, proporcional à altura.

ABB: árvore binária onde, para cada nó x , todo nó da subárvore esquerda de x tem *info* menor que $info(x)$, e todo nó da subárvore direita de x tem *info* maior que $info(x)$.

Árvores de busca binária (ABB)

Dicionário: tipo abstracto de dados para um conjunto com operações de **inserção**, **remoção** e **busca**.

ABBs servem para implementar dicionários e filas de prioridades.

Cada operação consome tempo, no pior caso, proporcional à altura.

ABB: árvore binária onde, para cada nó x , todo nó da subárvore esquerda de x tem *info* menor que $info(x)$, e todo nó da subárvore direita de x tem *info* maior que $info(x)$.

Percurso inordem de uma ABB visita os nós em ordem de *info*.

Busca em ABBs

BUSQUE (T, x)

1 **se** $T = \text{NIL}$ **ou** $\text{info}(T) = x$

2 **então devolva** T

3 **senão se** $x < \text{info}(T)$

4 **então devolva** $\text{BUSQUE}(\text{esq}(T), x)$

5 **senão devolva** $\text{BUSQUE}(\text{dir}(T), x)$

Busca em ABBs

BUSQUE (T, x)

- 1 **se** $T = \text{NIL}$ **ou** $\text{info}(T) = x$
- 2 **então devolva** T
- 3 **senão se** $x < \text{info}(T)$
- 4 **então devolva** $\text{BUSQUE}(\text{esq}(T), x)$
- 5 **senão devolva** $\text{BUSQUE}(\text{dir}(T), x)$

Versão iterativa:

BUSQUE (T, x)

- 1 $p \leftarrow T$
- 2 **enquanto** $p \neq \text{NIL}$ **e** $\text{info}(p) \neq x$ **faça**
- 3 **se** $x < \text{info}(p)$
- 4 **então** $p \leftarrow \text{esq}(p)$
- 5 **senão** $p \leftarrow \text{dir}(p)$
- 6 **devolva** p

Mais rotinas em ABBs

MÍNIMO (T) \triangleright supondo $T \neq \text{NIL}$

1 $p \leftarrow T$

2 **enquanto** $esq(p) \neq \text{NIL}$ **faça**

3 $p \leftarrow esq(p)$

4 **devolva** p

Mais rotinas em ABBs

MÍNIMO (T) \triangleright supondo $T \neq \text{NIL}$

- 1 $p \leftarrow T$
- 2 **enquanto** $esq(p) \neq \text{NIL}$ **faça**
- 3 $p \leftarrow esq(p)$
- 4 **devolva** p

SUCCESSOR (q) $\triangleright q$ é um nó da árvore

- 1 **se** $dir(q) \neq \text{NIL}$
- 2 **então devolva** $\text{MÍNIMO}(dir(q))$
- 3 **senão** $p \leftarrow pai(q)$
- 4 $ant \leftarrow q$
- 5 **enquanto** $p \neq \text{NIL}$ **e** $ant = dir(p)$ **faça**
- 6 $ant \leftarrow p$
- 7 $p \leftarrow pai(p)$
- 8 **devolva** p

Inserção em ABBs

INSIRA (T, q) $\triangleright q$ é a célula nova

```
1   $p \leftarrow T$ 
2   $ant \leftarrow \text{NIL}$ 
3   $x \leftarrow \text{info}(q)$ 
4  enquanto  $p \neq \text{NIL}$  faça
5       $ant \leftarrow p$ 
6      se  $x < \text{info}(p)$ 
7          então  $p \leftarrow \text{esq}(p)$ 
8          senão  $p \leftarrow \text{dir}(p)$ 
9  se  $ant = \text{NIL}$ 
10     então  $T \leftarrow q$ 
11     senão se  $x < \text{info}(ant)$ 
12         então  $\text{esq}(ant) \leftarrow q$ 
13         senão  $\text{dir}(ant) \leftarrow q$ 
14   $\text{pai}(q) \leftarrow ant$ 
```