CI1057: Algoritmos e Estruturas de Dados III
Árvores Binárias

Profa. Carmem Hara

Departamento de Informática/UFPR

11 de setembro de 2024

Tipos Abstratos de Dados

- **Tipo de Dado**
  conjunto de valores associado a um conjunto de operações
- **Tipo Abstrato de Dados (TAD)**
  é um tipo de dado que é acessado SOMENTE através de uma *interface* (conjunto de funções ou operações).
  - **Cliente do TAD:** programa que utiliza o TAD
  - **Implementação do TAD:** programa que especifica o tipo de dado
  - A interface é **opaca**: o cliente não pode ver a implementação através da interface

TAD Pilha (LIFO - *Last In, First Out*)

Uma **pilha** é um TAD composto por 2 operações básicas:

- *empilha*: insere um novo item
- *desempilha*: remove o item que foi inserido por último

Interface

- `void criaPilha( Pilha* )`
  cria pilha vazia
- `int pilhaVazia( Pilha )`
  retorna 1 se a pilha estiver vazia e zero, caso contrário
- `void empilha( Item, Pilha* )`
  empilha um item na pilha
- `Item desempilha( Pilha*, Item* )`
  remove o item no topo da pilha

Exercício

Utilize o TAD Pilha para avaliar uma expressão na notação Polonesa reversa.

Exemplo de expressão:

$$21\ 3\ *\ 11.36\ 89\ *\ 25.7\ 5.3\ *\ -\ /$$

TAD Fila (FIFO - *First In, First Out*)

Uma **fila** é um TAD composto por 2 operações básicas:

- *insere*: insere um novo item no final da fila
- *remove*: remove o primeiro item da fila (o primeiro que foi inserido)

Interface

- `void criaFila( Fila* )`:
  cria uma fila vazia
- `int filaVazia( Fila )`:
  retorna 1 se a filha estiver vazia e zero, caso contrário
- `void insereFila( Item, Fila*)`:
  insere um novo item no final da fila
- `void removeFila( Fila*, Item* )`:
  remove o primeiro item da fila

TAD Árvore Binária - Interface

- `void criaArv( ArvBin* )`:
  cria uma árvore vazia
- `int arvVazia( ArvBin )`:
  retorna 1 se a árvore estiver vazia e zero, caso contrário
- `ArvBin insereArv( Item , ArvBin )`:
  insere um novo item na árvore
- `void escreveArv( ArvBin )`:
  escreve a árvore
- `int alturaArv( ArvBin )`:
  retorna a altura da árvore
- `int contaNoArv( ArvBin )`:
  conta a quantidade de nodos da árvore
- `int arvCompleta( ArvBin )`:
  retorna 1 se a árvore está completa
- `void freeArv( ArvBin )`:
  libera toda a memória alocada para a árvore

Exercício

Utilize as TADs Pilha e ArvBin para percorrer a árvore em pré-ordem.

Entrada: 50 30 70 15 20 80 90 0
Saida: 50 30 15 80 77 90 100

Exercício

Utilize as TADs Fila e ArvBin para percorrer a árvore por nível.

Entrada: 50 30 70 15 20 80 90 0
Saida: 50 30 80 15 77 90 100

Referências

- Secoes 5.6, 5.7 (Sedgewick)