

Estruturas de Dados

lista de exercícios 23

1. Fila-med

Imagine que alguém queira utilizar uma estrutura de dados de fila com as duas operações adicionais descritas abaixo:

`getMed()`: retorna o elemento que se encontra exatamente no meio da fila

(i.e., se a fila tem n elementos, a função retorna o elemento na posição $\lceil n/2 \rceil$)

`putMed(x)`: insere o elemento x exatamente no meio da fila

A ideia, é claro, é que as duas operações sejam implementadas em tempo $O(1)$.

Descreva um tipo abstrato de dados (i.e., estrutura de dados e interface) que satisfaça essa restrição, e apresente a implementação das duas operações acima.

2. Pilha e fila

Faça programas que realizem as seguintes tarefas:

- Inverter a ordem dos elementos de uma pilha
- Inverter a ordem dos elementos de uma fila

3. Avaliador de expressões funcional

Considere uma lista de caracteres $C[1..n]$ contendo uma expressão funcional com números binários. Por exemplo,

C:

1	n
<code>mult(100, soma(quad(101), sub(1011, 11)))</code>	

Convertendo os números binários para decimal, e as funções para a notação matemática usual, nós obtemos a expressão

$$4 * (5^2 + (11 - 3))$$

Faça um programa que lê a lista $C[1..n]$ caracter a caracter, e retorna o valor da expressão que ela contém.

Nota: Você pode assumir que as únicas funções são: `mult`, `quad`, `soma` e `sub`.

4. Parênteses desbalanceados (OPCIONAL)

É comum cometer pequenos erros de digitação ao escrever uma expressão aritmética, de modo que os seus parênteses fiquem desbalanceados. Por exemplo,

$$4 + (3 * (5 + 2 () * 5$$

Faça um programa que calcule a menor quantidade de inversões de parênteses necessária para tornar a expressão balanceada, e reescreva a expressão após a devida correção.

5. Desafio

Você consegue implementar a operação abaixo no tipo abstrato Pilha-Min em tempo $O(\log n)$:

`deleteMin()`: remove o menor elemento da estrutura de dados