

Listas Lineares

Prof. Flavio B. Gonzaga
flavio.gonzaga@unifal-mg.edu.br
Universidade Federal de Alfenas
UNIFAL-MG

Sumário

- Listas lineares
- Alocação sequencial
 - Operações:
 - Inserção;
 - Remoção;
 - Pilhas
 - Inserção;
 - Remoção;

Operações

- Ambas as operações de inserção e remoção utilizam o procedimento de busca.
 - Na inserção o objetivo é evitar chaves repetidas.
 - Na remoção o objetivo é localizar o elemento a ser removido.

Inserção

- Retorna:

- -1 no caso de lista cheia.
- 0 no caso de elemento já existente.
- tamanho da lista no caso de inserção.

função *insere*(*no*)

se $n < m$ então

se $busca(no.chave) = 0$ então

$L[n + 1] := no$

$n := n + 1$

$insere := n$

senão $insere := 0$

senão $insere := -1$

Remoção

- Retorna:

- *nulo* no caso de lista vazia ou elemento inexistente.
- o nó caso contrário.

função *remove*(*x*)

remove := *nulo*

se $n \neq 0$ então

indice := *busca*(*x*)

se *indice* $\neq 0$ então

remove := *L*[*indice*]

para $i := \text{indice}, n - 1$ faça

$L[i] := L[i + 1]$

$n := n - 1$

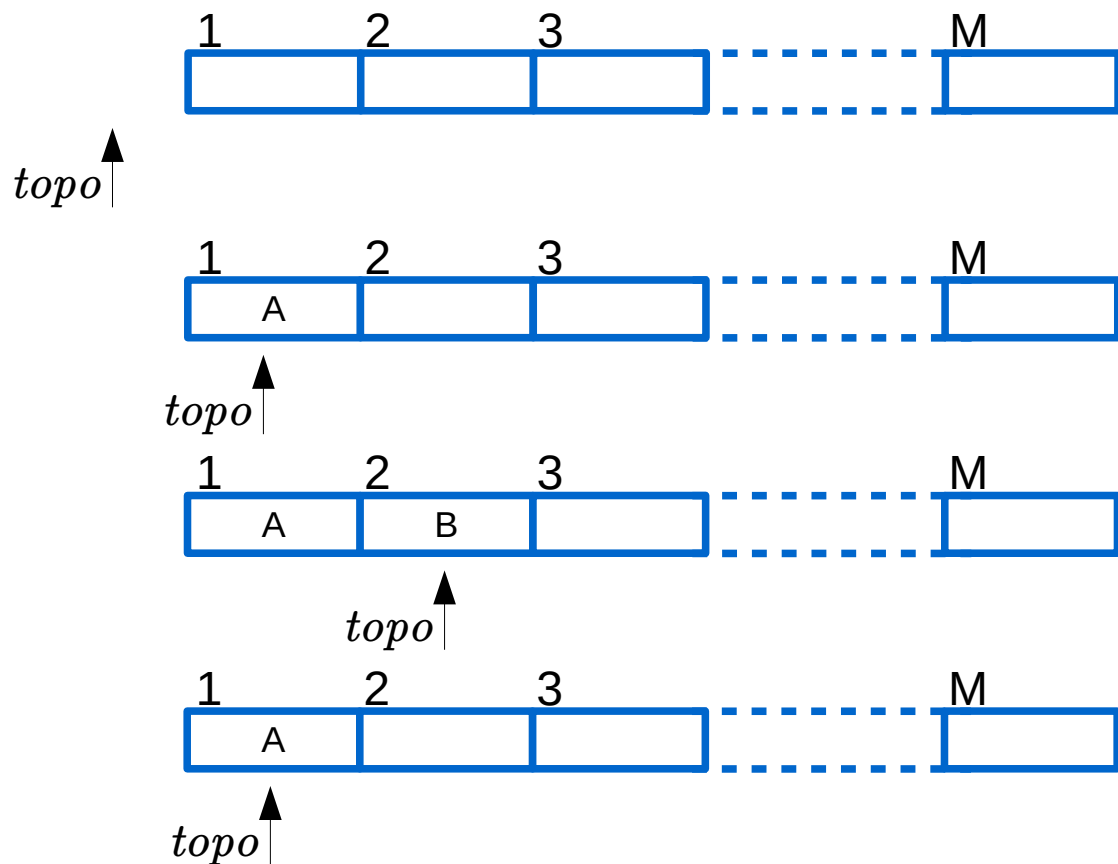
Observe que é possível efetuar o deslocamento do último elemento da lista para a posição vaga. Nesse caso entretanto, a sequência dos elementos será alterada.

Pilhas

- Em geral, o armazenamento sequencial de listas é empregado quando as estruturas, ao longo do tempo, sofrem poucas remoções e inserções.
- Em casos particulares de listas, esse armazenamento também é empregado.
- Nesse caso, a situação favorável é aquela em que inserções e remoções não acarretam movimentação de nós.
 - Tais como em deque, pilhas e filas.

- Na implementação de pilhas, faz-se necessário o armazenamento da localização do *topo* da mesma.

Pilhas



Inserção

- Retorna:
 - -1 no caso de pilha cheia.
 - tamanho da pilha no caso de inserção.

função *insere*(*no*)

insere := -1

se *topo* $\neq M$ então

topo := *topo* + 1

P[*topo*] := *no*

insere := *topo*

Remoção

- Retorna:
 - *nulo* no caso de pilha vazia.
 - o *nó* caso contrário.

função *remove*()

remove := *nulo*

se *topo* $\neq 0$ então

remove := *P*[*topo*]

topo := *topo* - 1

Referências Bibliográficas

- Estruturas de Dados e Seus Algoritmos. Szwarcfiter J. L.; Markenzon L.. 3a Edição. Editora LTC. 2010.
- Estruturas De Dados Usando C. Tenenbaum A. M.; Langsam Y.; Augenstein M. J.. 1a Edição. Editora Pearson. 1995.
- Introdução a Estruturas de Dados: Com Técnicas de Programação em C. Celes W.; Cerqueira R.; Rangel J.. 2a Edição. Editora Elsevier. 2017.