

Estrutura de Dados

Pilha

Por Juliano P. Menzen

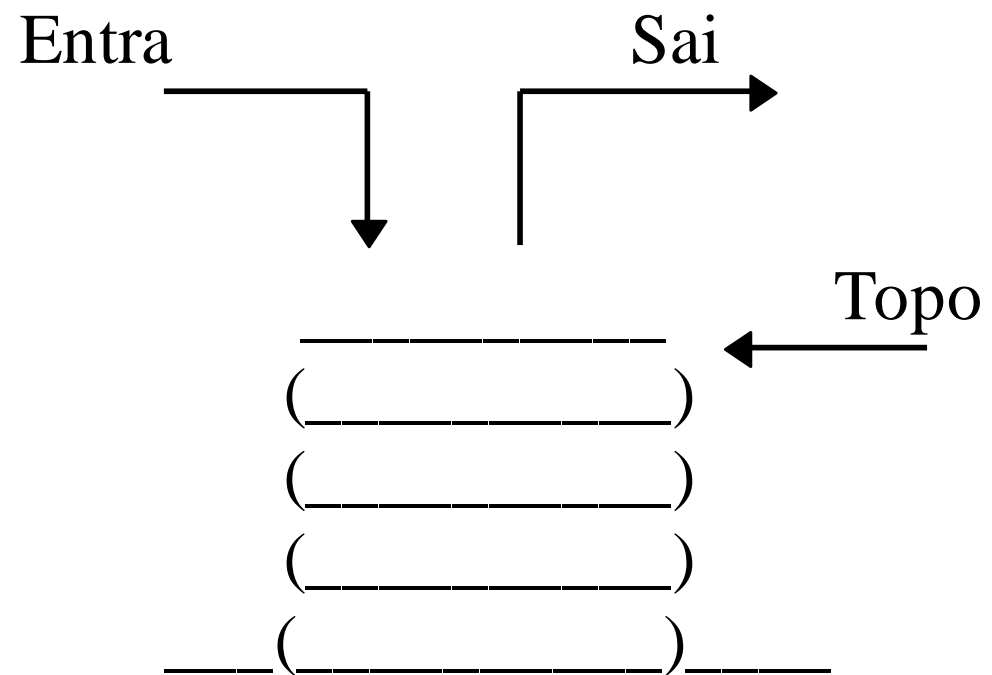
Sumário

- ▶ Conceitos;
- ▶ Operações básicas;
- ▶ Operações complementares;
- ▶ Implementação de pilha com Vetor;
- ▶ Implementação de pilha com Lista;
- ▶ Dúvidas;
- ▶ Exercícios;

Conceitos

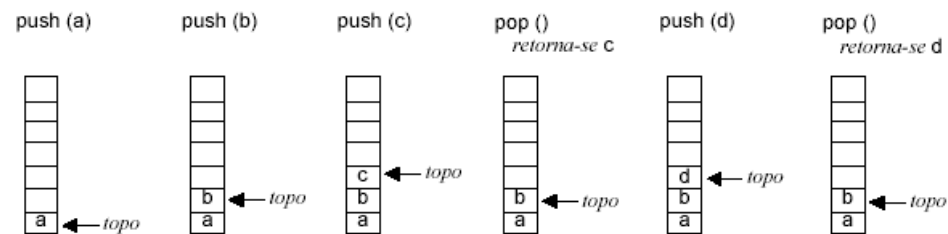
- ▶ Pilha é um tipo especial de lista linear em que todas as operações de inserção e remoção são realizadas numa mesma extremidade, denominada **topo**.
- ▶ Um novo elemento sempre é colocado no topo da mesma; e sempre que removemos um elemento este é retirado do topo da pilha;
- ▶ Devido a esta regra de acesso, os elementos são sempre removidos numa ordem inversa àquela em que foram inseridos, de modo que o último elemento que entra é exatamente o primeiro que sai;
- ▶ Por isso estas listas serem também denominadas de listas **LIFO** (Last-In/Frist-Out);
- ▶ Uma lista **LIFO** é uma **estrutura dinâmica**, ou seja, é uma coleção que pode aumentar e diminuir durante a sua existência;

Conceitos



Operações básicas

- ▶ Uma pilha suporta três operações básicas, tradicionalmente denominadas como:
 - ▶ **Top** : acessa o elemento posicionado no topo da pilha;
 - ▶ **Push** : insere um novo elemento no topo da pilha;
 - ▶ **Pop** : remove um elemento do topo da pilha.
- ▶ Estes nomes poderiam ser qualquer outro, pois temos que criar os algoritmos para que possamos manipular a pilha. (Insere, Retira, MostraTopo,...).



Operações Complementares

- ▶ **Init** : inicializa a pilha no estado “vazia”;
- ▶ **IsEmpty** : verifica se a pilha está vazia;
- ▶ **IsFull** : verifica se a pilha está cheia;

- ▶ A operação **Init(P)** tem como objetivo definir um estado inicial para a pilha **P**. Toda vez que criamos uma variável pilha, antes de qualquer coisa, devemos inicializá-la para garantir que não haverá nenhuma “sujeira” no local onde ela será “montada”;
- ▶ A operação **IsEmpty(P)**, que toma como argumento a pilha em que estamos interessados retorna um valor booleano verdadeiro se ela estiver vazia, sem nenhum elemento armazenado e falso caso contrário;
- ▶ A função **IsFull(P)** é usada para verificar se uma pilha está cheia, isto é , ela retorna verdadeiro somente quando não há mais espaço para armazenar nenhum elemento na pilha;

Operações Complementares

► Importante:

- Note que **IsFull(P)** não é o inverso de **IsEmpty(P)**. Quando **IsEmpty(P)** retorna falso, indicando que a pilha **P** não está vazia, não significa que ela se encontra cheia. Da mesma forma, quando **IsFull(P)** retorna falso, informando que a pilha **P** não está cheia, não quer dizer que ela esteja vazia. Uma pilha pode estar nem vazia, nem cheia.

Implementação de pilha com Vetor

- ▶ Pode ser utilizada quando o tamanho máximo da pilha é conhecido;
- ▶ Exemplo:
 - ▶ Ver arquivo “pilha com vetor.cpp”

Implementação de pilha com Lista

- ▶ Deve ser utilizada quando o número máximo de elementos que serão armazenados a pilha não é conhecido;
- ▶ Exemplo:
 - ▶ Ver arquivo “pilha com lista.cpp”

Dúvidas



Exercícios

- ▶ 1) Mostre a situação da pilha P, inicialmente vazia, após a execução de cada uma das operações a seguir:

Push(P,1);

Push(P,b);

Push(P,c);

Push(P,Top(P));

Push(P,Pop(P));

Pop(P):

Push(P,e);

Pop(p);

- ▶ 2) Escreva um programa para ler uma frase e imprima-la com a palavra invertidas. Exemplo: a frase “A pilha do gato” deve sair “A ahlip od otag”.
- ▶ 3) Estenda a funcionalidade da calculadora da apostila incluindo novos operadores (sugestão: # como raiz quadrada, ^ como exponenciação).