



Universidade Veiga de Almeida

Estrutura de Dados I

Parte 4 – Pilhas Estáticas.

Prof. Eng. Carlos A. Sicsú A. do Nascimento, DSc.



Estrutura de dados I



Pilhas Estáticas.

Estrutura de dados I

- Pilha é uma estrutura de dados em que as operações de inserção e remoção são realizadas em uma mesma extremidade, denominada TOPO.



Qual é o primeiro livro da pilha ?

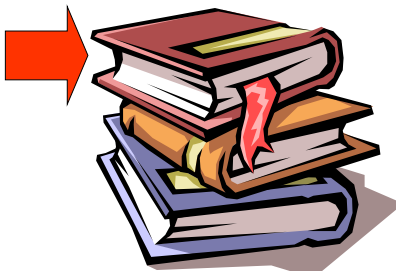
10/09/2019

Prof. Dr. Engº Carlos A. Sicsú A. do Nascimento

3

Estrutura de dados I

- Pilha é uma estrutura de dados em que as operações de inserção e remoção são realizadas em uma mesma extremidade, denominada TOPO.



Qual é o último livro da pilha ?

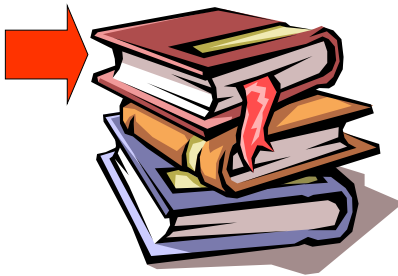
10/09/2019

Prof. Dr. Engº Carlos A. Sicsú A. do Nascimento

4

Estrutura de dados I

- Pilha é uma estrutura de dados em que as operações de inserção e remoção são realizadas em uma mesma extremidade, denominada TOPO.



Qual livro está no TOPO da pilha ?

10/09/2019

Prof. Dr. Engº Carlos A. Sicsú A. do Nascimento

5

Estrutura de dados I

Pilhas (L. I. F. O.)

A pilha também é conhecida como estrutura LIFO - Last In First Out

Último a entrar, primeiro a sair.

Dentre os elementos que ainda permanecem no conjunto, o primeiro elemento a ser retirado é o último que foi inserido.

10/09/2019

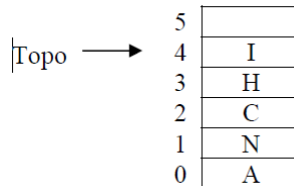
Prof. Dr. Engº Carlos A. Sicsú A. do Nascimento

6

Estrutura de dados I

Pilha (Stack)

É uma lista linear, em que todas as inserções e remoções de elementos só podem ser feitas numa extremidade chamada Topo.



10/09/2019

Prof. Dr. Engº Carlos A. Sicsú A. do Nascimento

7

Estrutura de dados I

Implementação (Estática)

A maneira mais simples de se representar uma pilha é através de um vetor de m elementos.

Digamos que este vetor chama-se P e seus elementos tem índices $0, 1, 2, \dots, m-1$.

O número máximo de elemento da pilha será m , o elemento do fundo será $P[0]$, o elemento seguinte $P[1]$ e assim por diante. Devido a este fato, esta representação é chamada seqüencial.

Associada a pilha tem-se uma variável chamada $Topo$, de tal forma que o elemento no topo da pilha será $P[Topo]$.

Pode-se convencionar que $Topo = -1$ indica uma pilha vazia.

10/09/2019

Prof. Dr. Engº Carlos A. Sicsú A. do Nascimento

8

Estrutura de dados I

Operações associadas com uma pilha:

- Inicializar uma pilha P vazia
- Inserir na pilha P
- Excluir da Pilha P
- Verificar se a pilha P está vazia

Exemplo de uso de Pilhas;

- processamento das chamadas de sub-rotinas e seus retornos
- simulação de sistemas
- sequencias bem-formadas
- Análise de expressões Pós fixas (Calculadores HP)

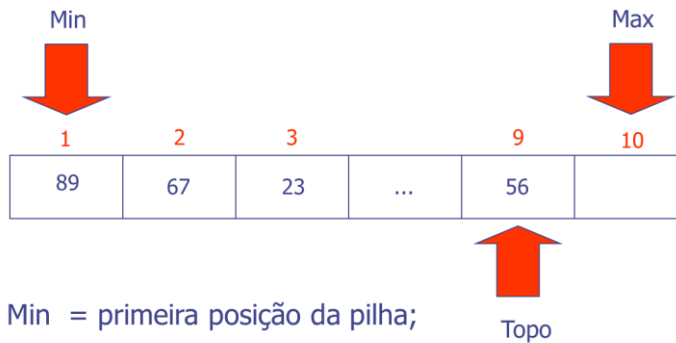
Estrutura de dados I

- A inserção e a retirada só podem ser realizadas no TOPO da pilha;

Somente DUAS operações:

- EMPILHAR (PUSH)
(inserir um novo elemento no topo da pilha).
- DESEMPILHAR (POP)
(remover o elemento do topo da pilha).

Pilhas Seqüenciais



Algoritmos para Pilha utilizando vetores

- Convenções:

Topo = -1 indica pilha vazia

Topo = TAM_VET - 1 indica pilha cheia

- Inicialização

Topo = -1

Estrutura de dados I

- INSERÇÃO

Se $\text{Topo} = \text{TAM_VET} - 1$ então

“PILHA CHEIA - INSERÇÃO IMPOSSÍVEL”

senão

$\text{Topo} = \text{Topo} + 1$

$P[\text{Topo}] = \text{Elemento}$

Fimse

Estrutura de dados I

- REMOÇÃO

Se $\text{Topo} = -1$ então

“PILHA VAZIA - REMOÇÃO IMPOSSÍVEL”

senão

$\text{Elemento} = P[\text{Topo}]$

$\text{Topo} = \text{Topo} - 1$

Fimse

Exemplo prático:

Implementação de pilhas com vetores em linguagem C - pilha de floats

```
#include <stdio.h>
#include <stdlib.h>

#define TAM_VET 10
#define VERD 1
#define FALSO 0
#define ERRO -1

void Inic_Pilha (int *Topo) {
    *Topo = -1;
}
```

10/09/2019

Prof. Dr. Engº Carlos A. Sicsú A. do Nascimento

15

```
int Pilha_Vazia(int *Topo) {
    if (*Topo == -1) return(VERD);
    return(FALSO);
}

int Pilha_Cheia(int *Topo) {
    if (*Topo == TAM_VET-1) return(VERD);
    return(FALSO);
}
```

10/09/2019

Prof. Dr. Engº Carlos A. Sicsú A. do Nascimento

16

Estrutura de dados I

```
int Empilha(float P[], int *Topo, float x) {
    if (*Topo == TAM_VET-1) {
        printf ( "Erro - Pilha cheia");
        return(ERRO);
    }
    else {
        (*Topo)++;
        P[*Topo] = x;
    }
    return(VERD);
}
```

Estrutura de dados I

```
int Desempilha(float P[], int *Topo, float *x) {
    if (*Topo == -1) {
        printf ( "Erro - Pilha vazia");
        return(ERRO);
    }
    else {
        *x = P[*Topo];
        (*Topo)--;
    }
    return(VERD);
}
```

Estrutura de dados I

```
int main(){
    float P[TAM_VET];
    int Topo;
    int ok, op=0;
    float dado;
    Inic_Pilha(&Topo);
    while ( op != 3 ) {
        printf ( "\n\nMenu\n1-Inserir\n2-Excluir\n3-Fim\nOpcao: ");
        scanf ("%d", &op);
        if ( op == 1 ) {
            printf ( "\nDigite um numero: ");
            scanf ("%f", &dado);
            Empilha(P,&Topo,dado);
        }
        else if ( op == 2 ) {
            ok = Desempilha(P,&Topo, &dado);
            if ( ok != ERRO)
                printf ( "\nElemento removido: %f ", dado);
        }
    }
    return 0;
}
```

10/09/2019

Prof. Dr. Engº Carlos A. Sicsú A. do Nascimento

19

Estrutura de dados I

- Exercício Prático 1 (Parte 4):

Você deve ter notado que as funções `Pilha_Cheia` e `Pilha_Vazia` não foram utilizadas no exemplo anterior, apesar de terem sido criadas. Reescreva o código incluindo o uso destas funções.

10/09/2019

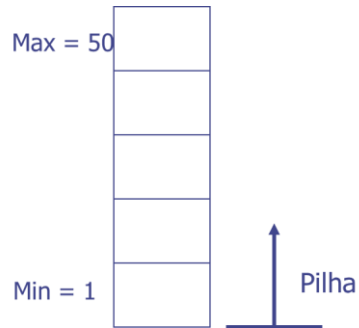
Prof. Dr. Engº Carlos A. Sicsú A. do Nascimento

20

Estrutura de dados I

- Exercício Prático 2 (Parte 4):

Suponha a seguinte estrutura de pilha sequencial (inteiros):



Escreva os procedimentos em C para empilhar e desempilhar a Pilha.

Estrutura de dados I

Obrigado pela atenção!

Fim da parte 4.