

# Aula 9 - Pilhas - Aplicações e exemplos práticos

Estruturas de Dados 2018/1

Prof. Diego Furtado Silva

# Balanceamento de parênteses

Dada uma sequência de parênteses (“abre” e “fecha”), verificar se eles formam uma **sequência válida**, ou seja:

- Cada parêntese aberto possui seu respectivo parêntese “fecha”
- Não há nenhum parêntese fechando uma expressão sem um respectivo “abre”

Exs:

((())) - válida

()()()()() - válida

( - inválida

() - inválida

# Balanceamento de parênteses

Conseguimos fazer essa verificação bem facilmente usando pilhas.

```
Tipo_pilha P // inicia vazia
```

```
para cada caracter c da sequência faça
```

```
    Se c == '('
```

```
        empilha(P,c)
```

```
    Senão
```

```
        Se pilha_vazia
```

```
            INVALIDO
```

```
        Senão
```

```
            desempilha(P)
```

```
Se conta(P) > 0
```

```
    INVALIDO
```

```
Senão
```

```
    VALIDO
```

Sim, o código vai estar disponível no AVA.

# Balanceamento de parênteses

O que acontece se eu misturar parênteses e chaves e/ou colchetes?

Fica como exercício. Lembrem-se que pode haver (ou não) uma hierarquia entre colchetes, chaves e parênteses.

# Verificação de expressões matemáticas

Dada uma expressão matemática, verificar se ela é válida.

Notação infixa: o **operador** é escrito **entre dois operandos**

Notação pós-fixa: primeiro escrevemos os **operandos e depois o operador**

Notação pré-fixa: o **operador** é escrito **antes dos operandos**

Infixa	Pós-fixa	Pré-fixa
$a + b$	$a b +$	$+ a b$
$a + b * c$	$a b c * +$	$+ a * b c$
$(a + b) * (c - d)$	$a b + c d - *$	$* + a b - c d$

# Verificação de expressões matemáticas

Verificação feita na notação pós-fixa é direta com pilha.

Vamos ao código

para cada elemento E da expressão faça

Se E é operando / número

empilha(P,E)

Senão //operador

V1 = desempilha(P)

V2 = desempilha(P)

empilha(V1 operador V2)

Se conta(P) > 1

INVALIDO

Senão

Resultado = topo(P)

# Verificação de expressões matemáticas

Verificação feita na notação pós-fixa é direta com pilha

Mas ninguém usa (naturalmente) essa notação. Então, como tornar a infixa (mais comumente usada) em pós-fixa?

Mais código =)

# Verificação de expressões matemáticas

para cada elemento E da expressão faça

Se E é operando / número

Imprime E

Senão se E é '('

empilha(P, E)

Senão se E ')'

imprime(desempilha(P)) até que topo(P) seja '('

Senão // operador

enquanto precedência(E) <= precedência(topo(P))

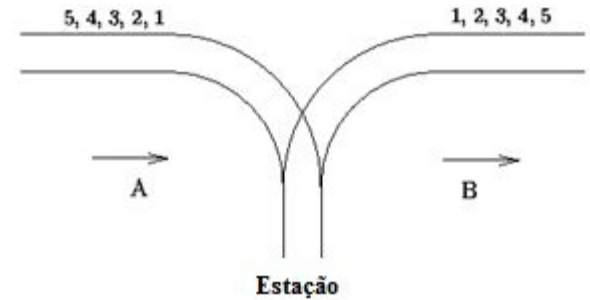
imprime(desempilha(P))

empilha(P, E)



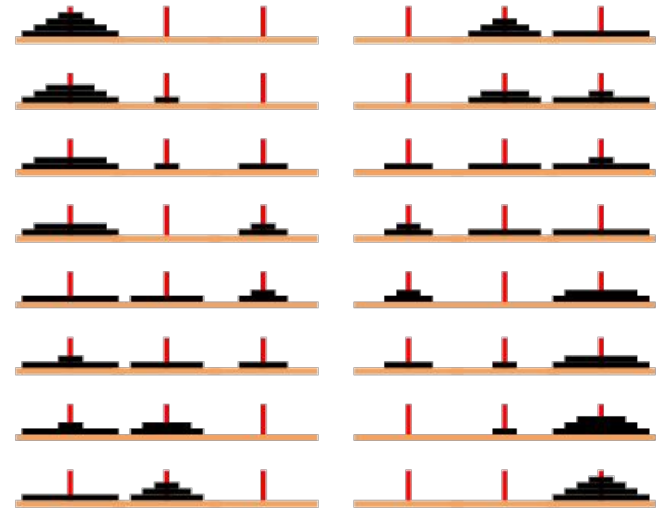
# Problema “Trilhos”

Numa estação com a configuração ao lado, verificar se é possível reordenar os vagões de um trem de acordo com a demanda.



# Torres de Hanoi

Exercício: resolver a torre de Hanoi usando pilhas



Fonte: Wolfram Alpha