

## Editor de texto: exercício

- Considere que um editor de texto representa os caracteres digitados como uma pilha, sendo que o último caractere lido fica no topo
- Alguns comandos apagam caracteres. Por exemplo, o *backspace* apaga o último caractere lido
- Alguns comandos apagam tudo o que já foi lido anteriormente
- Considere que, no seu editor, # representa *backspace* e @ indica “apagar tudo”
- Faça um programa que execute essas ações usando o TAD pilha

21

## Notação posfixa: exercício

- Avaliação de expressões aritméticas
  - Às vezes, na aritmética tradicional, faz-se necessário usar parênteses para dar o significado correto à expressão
    - $A*B-C/D \rightarrow (A*B)-(C/D)$
  - Notação polonesa (prefixa): operadores aparecem antes dos operandos e dispensam parênteses
    - $-*AB/CD$
  - Notação polonesa reversa (posfixa): operadores aparecem depois dos operandos
    - $AB*CD/-$

23

## Notação posfixa: exercício

- Interpretação da notação posfixa usando pilha
  - Empilha operandos até encontrar um operador
  - Retira os operandos, calcula e empilha o resultado
  - Até que se chegue ao final da expressão

24

## Notação posfixa: exercício

- $AB*CD/-$

A					
A	B				
A	B	*			
A*B					
A*B	C				
A*B	C	D			
A*B	C	D	/		
A*B	C/D				
A*B	C/D	-			
A*B-C/D					

## Exercício

- Implemente uma função que calcule o valor de uma expressão posfixa passada por parâmetro utilizando o TAD pilha
- Adicionar uma rotina ao TAD para verificar se duas pilhas são iguais
- Adicionar uma rotina ao TAD para inverter a posição dos elementos de uma pilha

26

## Exercício – resposta algorítmica

```

função valor(E: expressão): retorna real;
declare x real;
declare P pilha;
início
  Create(P)
  enquanto não acabou(E) faça
    início
      x=proxsimb(E);
      se x é operando então Push(P,x)
      senão início
        remove operandos; {dois pops, em geral}
        calcula o resultado da operação;
        empilhe resultado; {push}
      fim
    fim
  fim
  valor=Top(P);
fim
    
```

27