

3ª Prova de Algoritmos e Estruturas de Dados I

19/06/2019

O que será avaliado? Especialmente nesta prova, a demonstração que entendeu o conceito de tipos abstratos de dados, mas também a questão da modularidade: uso de funções e procedimentos, passagem de parâmetros e uso de variáveis locais. Mas também contam: a clareza, a lógica, a criatividade, a sintaxe, o uso correto dos comandos, a correta declaração dos tipos, os nomes das variáveis, a indentação e o uso equilibrado de comentários no código.

QUESTÃO 1: (40 pontos)

Considere o tipo abstrato de dados Pilha e os protótipos das funções e procedimentos:

```
CONST MAX = 100;
TYPE
    elemento = longint;
    pilha = record
        ini: longint;
        v: array [1..MAX] of elemento;
    end;

procedure inicializa_pilha (var p: pilha);
(* inicializa a estrutura *)

function pilha_vazia (p: pilha): boolean;
(* retorna true se a pilha esta vazia *)

procedure empilha (x: elemento; var p: pilha);
(* insere x no inicio da pilha *)

function desempilha (var p: pilha): elemento;
(* retorna o elemento do inicio da pilha e o remove dela *)

function topo (p: pilha): elemento;
(* retorna o elemento do inicio da pilha, sem remove-lo *)
```

Implemente um programa em *Pascal* que receba como entrada uma lista de números inteiros terminada em zero. O zero não deve ser processado e apenas indica que a entrada de dados terminou. Seu programa deve imprimir SIM caso a lista de números tenha pares bem formados de números de mesmo valor absoluto, sendo que o positivo ocorre sempre antes do seu respectivo negativo. Seu programa deve imprimir NAO em caso contrário. Apenas a título de exemplo, caso os números fossem todos 1 e -1, isto poderia ser visto como parenteses abertos (1) e parênteses fechados (-1), representando uma expressão de parênteses corretamente balanceada. Não é necessário escrever os códigos das funções e procedimentos do TAD Pilha, considere que eles estão corretamente implementados em uma biblioteca. Apenas resolva o problema *usando* o TAD Pilha.

Exemplo 1:

```
1 2 3 4 -4 -3 5 6 -6 -5 -2 -1
SIM
```

Exemplo 2:

```
1 2 3 -1 -2 -3
NAO
```

QUESTÃO 2: (60 pontos)

Considere que a partir de uma matriz podemos definir uma *ampulheta* como sendo feita de 7 células da seguinte forma:

```
A B C
  D
E F G
```

Por exemplo, a matriz seguinte:

```
2 3 0 0 0
0 1 0 0 0
1 1 1 0 0
0 0 2 4 4
0 0 0 2 0
```

possui as seguintes 9 ampulhetas:

```
2 3 0   3 0 0   0 0 0
  1       0       0
1 1 1   1 1 0   1 0 0
```

```
0 1 0   1 0 0   0 0 0
  1       1       0
0 0 2   0 2 4   2 4 4
```

```
1 1 1   1 1 0   1 0 0
  0       2       4
0 0 0   0 0 2   0 2 0
```

a soma dos elementos de cada uma das ampulhetas é:

9, 5, 1, 4, 8, 10, 3, 6 e 7

Faça um programa em *Pascal* que leia uma matriz de números inteiros de dimensão quadrada $0 \leq N \leq 100$, e que imprima o maior valor da soma dos elementos dentre todas as ampulhetas desta matriz.

Dado o exemplo acima a saída esperada do programa é:

10

Não esqueça que o uso de funções e procedimentos é fundamental nesta prova.