

2018_1 - ALGORITMOS E ESTRUTURAS DE DADOS II - TN

PAINEL > MINHAS TURMAS > 2018 1 - ALGORITMOS E ESTRUTURAS DE DADOS II - TN

> VPL CHALLENGE 02: EXERCICIOS DE LISTAS/PILHAS - PROVA 2 > P2.5 - REORGANIZAÇÃO DE PILHAS

Descrição

[Visualizar envios](#)

P2.5 - Reorganização de pilhas

Data de entrega: terça, 1 Mai 2018, 23:55

Arquivos requeridos: pilha.h, pilha.c, main.c ([Baixar](#))

Tipo de trabalho: Trabalho individual

Dado duas pilhas ordenadas em ordem decrescente, ou seja, o maior elemento se encontra no topo da pilha. Crie uma terceira pilha que esteja ordenada de forma crescente, ou seja, o menor elemento no topo da pilha, sendo esta uma combinação das outras duas pilhas.

As pilhas de entrada não necessariamente têm o mesmo tamanho, são dadas na ordem correta e contém elementos diferentes.

Exemplo:

Primeiros dois números da entrada: 6 e 4, logo, a primeira pilha tem tamanho 6 e a segunda tamanho 4 (Use esta informação para ajudar na leitura da entrada)

Pilha 1, o elemento 9 ficara no topo da pilha:

1 2 3 4 6 9

Pilha 2, o elemento 10 ficara no topo da pilha:

5 7 8 10

Pilha resultante, o elemento 1 ficou no topo e é o primeiro a ser impresso:

1 2 3 4 5 6 7 8 9 10

Exemplo de arquivo pilha.h:

```
#ifndef PILHA_H
#define PILHA_H

typedef struct node {
    int valor;
    struct node *next;
} node_t;

typedef struct {
    int num_elementos;
    node_t *inicio;
} pilha_t;

pilha_t *cria_pilha();
void push(pilha_t *pilha, int valor);
int pop(pilha_t *pilha);
void combinacao_pilhas(pilha_t *pilha_A, pilha_t *pilha_B, pilha_t *pilha_AB);

#endif
```

A entrada é a seguinte:

PRIMEIROS DOIS NÚMEROS - Tamanho das duas pilhas.

NUMEROS CONSECUTIVOS - Valores a serem inseridos na ordem dada na pilha

A saída deve ser a combinação de ambas as pilhas em ordem crescente.

Exemplo de arquivo main:

```
#include "pilha.h"

int main(void) {
    // Aloca Lista
    // EXEMPLOS DE LEITURA DA ENTRADA
    while(scanf("%d", VARIABEL) != '\n')
        while(scanf("%d", VARIABEL) != EOF)

    // Insere elementos
    // Imprime saída
    return 0;
}
```

[VPL](#)

[◀ P2.3.1 Palindromo V2](#)

Seguir para...

[P2.6 - Minimo pilha ▶](#)