

# 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** > **P2.3.1 PALINDROMO V2**

Descrição

[Visualizar envios](#)

## P2.3.1 Palindromo V2

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

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

**Tipo de trabalho:** Trabalho individual

Crie um algoritmo que checa se uma lista é um palíndromo utilizando uma pilha. Ou seja, seu palíndromo será inserido em uma lista **simplesmente encadeada**. Você deve usar uma pilha para checar se ele é um palíndromo ou não.

A estrutura da lista e da pilha devem salvar um ponteiro para a próxima célula e um char. Logo, cada célula da lista e pilha representam uma letra de uma palavra.

O algoritmo deve conter funções que:

1. Crie uma lista vazia;
2. Crie um pilha vazia;
3. Insira um novo elemento em uma lista, neste caso, insira a letra na variável char da estrutura;
4. Verifica se a lista é um palíndromo utilizando uma pilha de mesmo tamanho da lista.

Dica: Um palíndromo, é uma *string* que pode ser lida da esquerda para a direita da mesma forma que é lida da direita para a esquerda. Ou seja, as letras são espelhadas até certo ponto.

Exemplo lista.h:

```
#ifndef LISTA_H
#define LISTA_H

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

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

lista_t *cria_lista();
void insere_valor(lista_t *lista, char valor);
int palindromo(lista_t *lista, pilha_t *pilha);

#endif
```

Exemplo pilha.h

```

#ifndef PILHA_H
#define PILHA_H

typedef struct node {
    char 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);
#endif

```

Exemplo de main;

```

#include "lista.h"
#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;
}

```

A entrada é dada por N letras que devem ser inseridas nas células da lista. A saída deve sinalizar se a lista é ou não um palíndromo:

Caso a lista seja um palíndromo retorne:

1

Caso a lista não seja um palíndromo retorne:

0

## Arquivos requeridos

lista.h

lista.c

pilha.h

pilha.c

main.c

```

1 |
2 | // Para ler a entrada insira um espaço no scanf(" %c", variavel)

```

[VPL](#)

◀ [P2.3 - Palindromo](#)

Seguir para...

[P2.5 - Reorganização de pilhas](#) ▶