

1
2
3
4
5
6
7
8
9
10
11
12
13
14

Exercícios Propostos III: Exercício 3

1

2

3

4

5 Exercício 3. Escreva um programa que leia uma palavra do
6 teclado e determine se é um palíndromo. Observação: Palíndromo
7 é uma palavra, frase ou número que permanece igual quando lida
8 de trás para diante.

9

10

11

12

13

14

1
2
3
4
5
6
7
8
9
10
11
12
13
14

Alternativa A

```
1 #include <stdio.h>
2 #include <string.h>
3 #include <locale.h>
4 int main(){
5     setlocale(LC_ALL, "Portuguese");
6     char palavra[30];
7
8     printf("Digite uma palavra: ");
9     fgets(palavra, 30, stdin);
10    palavra[ strcspn( palavra, "\n" ) ] = '\0';
11
12    int i = 0;
13    int cont = strlen(palavra);
14    cont--;
```

```
1  while(cont - 1 >= i){
2      if(tolower(palavra[i]) != tolower(palavra[cont])){
3          errado++;
4      }
5      i++;
6      cont--;
7  }
8  if(errado == 0){
9      printf("\nA palavra %s é palíndroma!\n", palavra);
10 } else{
11     printf("\nA palavra %s não é palíndroma!\n", palavra);
12 }
13 return 0;
14 }
```

1
2
3
4
5
6
7
8
9
10
11
12
13
14

Alternativa B

```
1 #include<stdio.h>
2 #include<string.h>
3 #include <locale.h>
4 void main()
5 {
6     setlocale(LC_ALL, "Portuguese");
7     char palavra[20];
8     char inversa[20];
9     printf("Digite uma palavra: ");
10    fgets(palavra, 20, stdin);
11    palavra[ strcspn( palavra, "\n" ) ] = '\0';
12    for(int i = 0; palavra[i]; i++){
13        palavra[i] = tolower(palavra[i]);
14    }
```

```
1  strcpy(inversa, palavra);
2  strrev(inversa);
3
4  int comparacao = 0;
5  comparacao = strcmp(palavra, inversa) == 0;
6
7  if(comparacao){
8      printf("\n%s é um palíndromo\n", palavra);
9  }
10 else{
11     printf("\n%s NÃO é um palíndromo\n", palavra);
12 }
13
14
```


EXECUTANDO

1

2 **Funciona:**

3

4 Digite uma palavra: Mussum

5

6 A palavra Mussum é palíndroma!

7

8

9

10

Não funciona:

11

12 Digite uma palavra: amora

13

14 A palavra amora não é palíndroma!

15

1
2
3
4
5
6
7
8
9
10
11
12
13
14

Obrigado pela atenção!