

CI1055: Algoritmos e Estruturas de Dados I

Profs. Drs. Marcos Castilho, Carmem Hara e Bruno Müller Jr

Departamento de Informática/UFPR

21 de outubro de 2021

Aplicação das técnicas elementares

- ▶ Fatorial
- ▶ Fibonacci revisado
- ▶ (*) Palíndromos

Palíndromos

- ▶ Palíndromos são lidos da direita para a esquerda da mesma maneira que da esquerda para a direita. Por exemplo o número 12321 é palíndromo, enquanto que 123 não é;
- ▶ Método: separar os dígitos do número, reconstruir o número ao contrário, comparar com o número original.
- ▶ generalização do problema de se inverter um número de três dígitos. Dicas:

$$d_n d_{n-1} \dots d_2 d_1 d_0 \Rightarrow d_0 d_1 d_2 \dots d_{n-1} d_n$$

$$(12345) \text{ div } 10 = 1234 \quad (1)$$

$$(12345) \text{ mod } 10 = 5 \quad (2)$$

$$(1234) * 10 + 5 = 12345 \quad (3)$$

Palíndromos

- ▶ Palíndromos são lidos da direita para a esquerda da mesma maneira que da esquerda para a direita. Por exemplo o número 12321 é palíndromo, enquanto que 123 não é;
- ▶ Método: separar os dígitos do número, reconstruir o número ao contrário, comparar com o número original.
- ▶ generalização do problema de se inverter um número de três dígitos. Dicas:

$$d_n d_{n-1} \dots d_2 d_1 d_0 \Rightarrow d_0 d_1 d_2 \dots d_{n-1} d_n$$

$$(12345) \text{ div } 10 = 1234 \quad (1)$$

$$(12345) \text{ mod } 10 = 5 \quad (2)$$

$$(1234) * 10 + 5 = 12345 \quad (3)$$

Palíndromos

- ▶ Palíndromos são lidos da direita para a esquerda da mesma maneira que da esquerda para a direita. Por exemplo o número 12321 é palíndromo, enquanto que 123 não é;
- ▶ Método: separar os dígitos do número, reconstruir o número ao contrário, comparar com o número original.
- ▶ generalização do problema de se inverter um número de três dígitos. Dicas:

$$d_n d_{n-1} \dots d_2 d_1 d_0 \Rightarrow d_0 d_1 d_2 \dots d_{n-1} d_n$$

$$(12345) \text{ div } 10 = 1234 \quad (1)$$

$$(12345) \text{ mod } 10 = 5 \quad (2)$$

$$(1234) * 10 + 5 = 12345 \quad (3)$$

Palíndromos

Problema: Imprimir todos os números palíndromos entre 1 e 1000.

```
program todospalindromos ;
const max=1000;
var i , invertido , n: integer ;
begin
  i:= 1;
  while i <= max do (* laço que controla os numeros entre 1 e max *)
  begin
    invertido:= 0; (* inicializa acumulador *)
    n:= i ;
    while n > 0 do
    begin
      invertido:= invertido*10 + n mod 10;
      n:= n div 10;
    end;
    (* imprime se for palindromo , senao nao faz nada *)
    if invertido = i then
      writeln ( i ) ;
    i:= i + 1;
  end;
end.
```

Palíndromos

Problema: Gerar todos os números palíndromos entre 1 e 1000.

- ▶ Uma outra abordagem ao problema baseia-se no fato de o maior palíndromo ocupar 3 dígitos:
 1. gerar todos os palíndromos de um dígito (muito fácil);
 2. gerar todos os palíndromos de dois dígitos (muito fácil);
 3. gerar todos os palíndromos de três dígitos. Dica:

$$d[0 - 9]d$$

Palíndromos

Problema: Gerar todos os números palíndromos entre 1 e 1000.

- ▶ Uma outra abordagem ao problema baseia-se no fato de o maior palíndromo ocupar 3 dígitos:
 1. gerar todos os palíndromos de um dígito (muito fácil);
 2. gerar todos os palíndromos de dois dígitos (muito fácil);
 3. gerar todos os palíndromos de três dígitos. Dica:

$$d[0 - 9]d$$

Palíndromos

Problema: Gerar todos os números palíndromos entre 1 e 1000.

- ▶ Uma outra abordagem ao problema baseia-se no fato de o maior palíndromo ocupar 3 dígitos:
 1. gerar todos os palíndromos de um dígito (muito fácil);
 2. gerar todos os palíndromos de dois dígitos (muito fácil);
 3. gerar todos os palíndromos de três dígitos. Dica:

$$d[0 - 9]d$$

Palíndromos

Problema: Gerar todos os números palíndromos entre 1 e 1000.

- ▶ Uma outra abordagem ao problema baseia-se no fato de o maior palíndromo ocupar 3 dígitos:
 1. gerar todos os palíndromos de um dígito (muito fácil);
 2. gerar todos os palíndromos de dois dígitos (muito fácil);
 3. gerar todos os palíndromos de três dígitos. Dica:

$$d[0 - 9]d$$

Palíndromos

Problema: Gerar todos os números palíndromos entre 1 e 1000.

```
program gerandopalindromos ;
var i , j , pal : integer ;
begin
  ... (* gerando todos de um digito *)
  pal:= 11; (* gerando todos de 2 digitos *)
  i:= 2;
  while i <= 9 do
  begin
    writeln (pal) ;
    pal:= i * 11;
    i:= i + 1;
  end;
  i:= 1; (* ----- *)
  while i <= 9 do (* gerando todos os de 3 digitos *)
  begin
    j:= 0;
    while j <= 9 do
    begin
      pal:= i*100 + j*10 + i ;
      writeln (pal) ;
      j:= j + 1;
    end;
    i:= i + 1;
  end;
end.
```

Fim

- ▶ este material está no livro no capítulo 7, seção 7.7

Licença

- ▶ Slides feitos em \LaTeX usando beamer
- ▶ Licença

Creative Commons Atribuição-Uso Não-Comercial-Vedada a Criação de Obras Derivadas 2.5 Brasil License.<http://creativecommons.org/licenses/by-nc-nd/2.5/br/>

Creative Commons Atribuição-Uso Não-Comercial-Vedada a Criação de Obras Derivadas 2.5 Brasil License.<http://creativecommons.org/licenses/by-nc-nd/2.5/br/>