De volta aos algoritmos

Alexsandro Santos Soares
prof.asoares@gmail.com

Universidade Federal de Uberlândia
Faculdade de Computação

Exercício 1

Na matemática, o número de Euler **e**, denominado em homenagem ao matemático suíço Leonhard Euler, é a base dos logaritmos naturais. Seu valor é aproximadamente

$$e = 2,718\ 281\ 828\ 459\ 045\ 235\ 360\ 287.$$

O número e pode ser representado e calculado como a soma da seguinte série infinita:

$$e = \sum_{n=0}^{\infty} \frac{1}{n!} = \frac{1}{0!} + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \frac{1}{4!} + \cdots$$

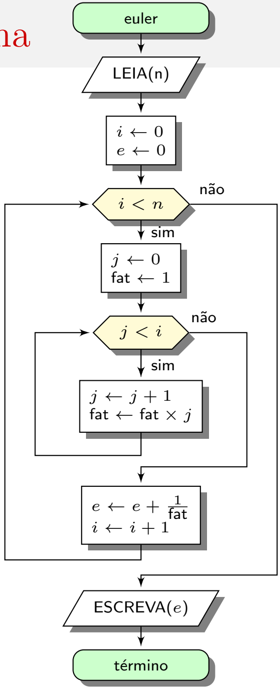
Escreva um algoritmo que leia um número natural **n** e depois calcule e imprima o valor do número de Euler aproximado com n termos.

1. Elabore um conjunto significativo de casos de testes.
2. Expresse o algoritmo na forma de fluxograma e na de pseudocódigo.
3. Teste o algoritmo usando tabela de rastreamento e os dados do item 1.

Algoritmo 1: Número de Euler

```
leia o número natural n
i ← 0                                  // contador de iterações do número de euler
e ← 0                                  // acumula o valor de e
enquanto i < n faça
    j ← 0                              // contador de iterações do fatorial
    fat ← 1                            // acumula o valor do fatorial
    enquanto j < i faça
        j ← j + 1
        fat ← fat × j
    fim enqto
    e ← e + 1/fat
    i ← i + 1
fim enqto
escreva "e =", e
```

Solução – fluxograma



Exercício 2

Faça um algoritmo que leia um número natural n, indicativo de quantos números naturais serão lidos a seguir. Depois, para cada número lido, o algoritmo deve imprimir o valor lido e o fatorial deste valor.

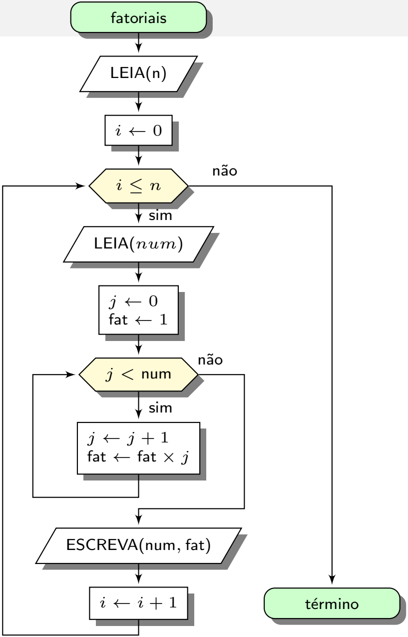
Observação: A partir de agora quando se pedir um algoritmo ou programa, o ítens abaixo serão sempre requisitados e não serão mais mostrados no enunciado.

1. Elabore um conjunto significativo de casos de testes.
2. Expresse o algoritmo na forma de fluxograma e na de pseudocódigo.
3. Teste o algoritmo usando tabela de rastreamento e os dados do item 1.

Algoritmo 2: Tabela de fatoriais

```
leia o número natural n                    // quantidade de números a serem lidos
i ← 0                                      // contador de iterações
enquanto i ≤ n faça
    leia o número natural num
    j ← 0                                  // contador de iterações do fatorial
    fat ← 1                                // acumula o valor do fatorial
    enquanto j < num faça
        j ← j + 1
        fat ← fat × j
    fim enqto
    escreva num, "   ", fat
    i ← i + 1
fim enqto
```

Fluxograma


fatoriais
LEIA(n)
$i \leftarrow 0$
$i \leq n$
não
sim
LEIA(num)
$j \leftarrow 0$
fat $\leftarrow 1$
$j <$ num
não
sim
$j \leftarrow j + 1$
fat $\leftarrow$ fat $\times j$
ESCREVA(num, fat)
$i \leftarrow i + 1$
término

Exercício 3

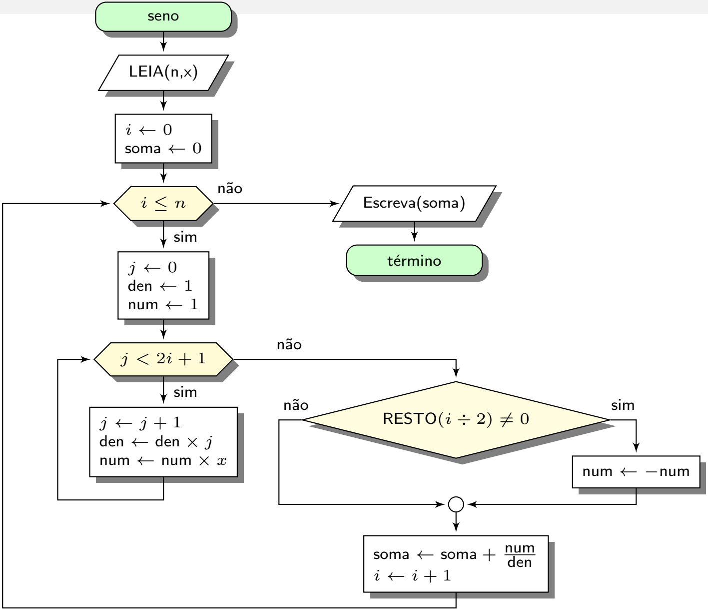
A função trigonométrica *seno(x)* pode ser calculada usando a fórmula a seguir

$$\text{seno(x)} = \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+1)!} x^{2n+1} = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots$$

Escreva um algoritmo que leia um número natural **n** e um número real **x** e depois calcule e imprima o valor do *seno(x)*, aproximado com n termos.

```
leia o número natural n                          // número de termo
leia o número real x                             // argumento do seno
i ← 0                                            // contador de iterações
soma ← 0                                         // acumula o valor do seno
enquanto i ≤ n faça
    j ← 0                                        // contador de iterações do fatorial
    denominador ← 1                              // acumula o valor de (2n + 1)!
    numerador ← 1                                // acumula o valor de x^(2n+1)
    enquanto j < 2i + 1 faça
        j ← j + 1
        denominador ← denominador × j
        numerador ← numerador × x
    fim enqto
    se RESTO(i ÷ 2) ≠ 0 então                    // ordem do termo é ímpar
        numerador ← −numerador
    fim se
    soma ← soma + numerador / denominador
    i ← i + 1
fim enqto
escreva "seno(", x, ")=", soma
```

Fluxograma


seno
LEIA(n,x)
$i \leftarrow 0$
soma $\leftarrow 0$
$i \leq n$
não
Escreva(soma)
término
sim
$j \leftarrow 0$
den $\leftarrow 1$
num $\leftarrow 1$
$j < 2i + 1$
não
sim
$j \leftarrow j + 1$
den $\leftarrow$ den $\times j$
num $\leftarrow$ num $\times x$
não
RESTO$(i \div 2) \neq 0$
sim
num $\leftarrow -$num
soma $\leftarrow$ soma $+ \frac{\text{num}}{\text{den}}$
$i \leftarrow i + 1$

Exercício 4

A sequência de Fibonacci é uma sequência de números inteiros, começando normalmente por 0 e 1, na qual, cada termo subsequente corresponde à soma dos dois anteriores. A sequência recebeu o nome do matemático italiano Leonardo de Pisa, mais conhecido por Fibonacci.

Abaixo estão os primeiros números de Fibonacci

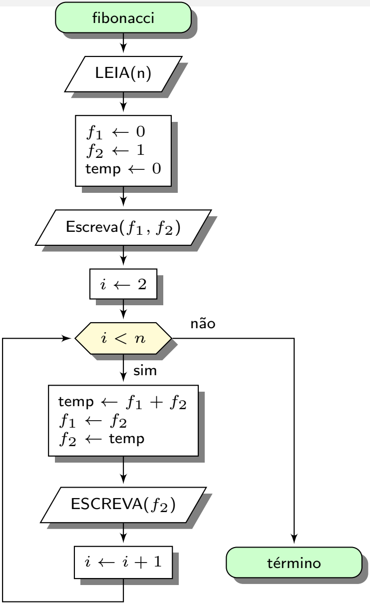
$$0,\ 1,\ 1,\ 2,\ 3,\ 5,\ 8,\ 13,\ 21,\ 34,\ 55,\ 89,\ 144,\ 233,\ 377,\ 610,\ 987,\ 1597,\ 2584, \ldots$$

Escreva um algoritmo que leia um número natural **n** e depois calcule e imprima os n primeiros números de Fibonacci.

Algoritmo 3: Fibonacci

leia o número natural n
$f_1 \leftarrow 0$ `/* o penúltimo número de Fibonacci calculado */`
$f_2 \leftarrow 1$ `/* o último número de Fibonacci calculado */`
temp $\leftarrow 0$ `/* temporário para guardar a soma */`
escreva f_1
escreva f_2
$i \leftarrow 2$ `/* contador de iterações */`
enquanto $i < n$ **faça**
    temp $\leftarrow f_1 + f_2$
    $f_1 \leftarrow f_2$
    $f_2 \leftarrow$ temp
    escreva f_2
    $i \leftarrow i + 1$
fim enqto

Fluxograma

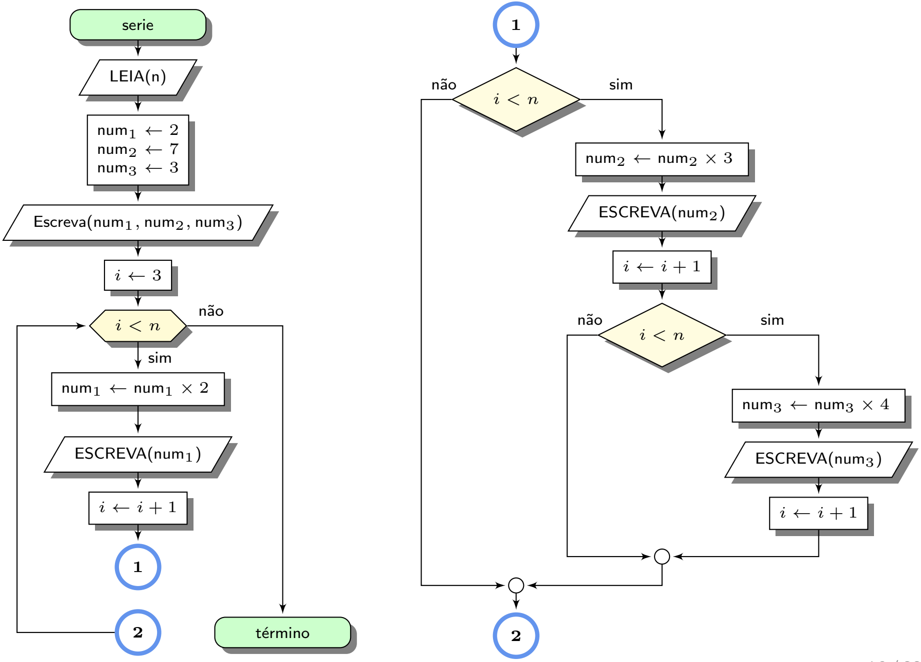

fibonacci
LEIA(n)
$f_1 \leftarrow 0$
$f_2 \leftarrow 1$
temp $\leftarrow 0$
Escreva(f_1, f_2)
$i \leftarrow 2$
$i < n$
não
sim
temp $\leftarrow f_1 + f_2$
$f_1 \leftarrow f_2$
$f_2 \leftarrow$ temp
ESCREVA(f_2)
$i \leftarrow i + 1$
término

Exercício 5

Escreva um algoritmo que leia um número natural **n** e depois calcule e imprima os n primeiros termos da série a seguir

$$2,\ 7,\ 3,\ 4,\ 21,\ 12,\ 8,\ 63,\ 48,\ 16,\ 189,\ 192,\ 32,\ 567,\ 768,\ \ldots$$

```
leia o número natural n
num₁ ← 2
num₂ ← 7
num₃ ← 3
escreva num₁, num₂, num₃
i ← 3                                          /* contador de iterações */
enquanto i < n faça
    num₁ ← num₁ × 2
    escreva num₁
    i ← i + 1
    se i < n então
        num₂ ← num₂ × 3
        escreva num₂
        i ← i + 1
        se i < n então
            num₃ ← num₃ × 4
            escreva num₃
            i ← i + 1
        fim se
    fim se
fim enqto
```


serie
LEIA(n)
$num_1 \leftarrow 2$
$num_2 \leftarrow 7$
$num_3 \leftarrow 3$
Escreva(num_1, num_2, num_3)
$i \leftarrow 3$
$i < n$
não
sim
$num_1 \leftarrow num_1 \times 2$
ESCREVA(num_1)
$i \leftarrow i + 1$
1
2
término
1
não
$i < n$
sim
$num_2 \leftarrow num_2 \times 3$
ESCREVA(num_2)
$i \leftarrow i + 1$
não
$i < n$
sim
$num_3 \leftarrow num_3 \times 4$
ESCREVA(num_3)
$i \leftarrow i + 1$
2

Exercício 6

Escreva um algoritmo que leia um número natural $n > 1$, verifique se o número fornecido é primo e mostre mensagens adequadas para cada caso.

Um número é primo se ele for maior que 1 e for divisível apenas por 1 ou por ele mesmo.

Algoritmo 4: Primo

```
leia o número natural n
ndivisores ← 0                                  // número de divisores
i ← 1                                           // contador de iterações
enquanto i ≤ n faça
    se RESTO(n ÷ i) = 0 então                   // encontramos um divisor
        ndivisores ← ndivisores + 1
    fim se
    i ← i + 1
fim enqto
se ndivisores > 2 então
    escreva "Não é primo"
senão
    escreva "É primo"
fim se
```

Exercício 7

Escreva um algoritmo que leia um número não determinado de pares de valores $[m, n]$, ambos naturais e positivos, um par de cada vez, e que calcule e mostre a soma de todos os números naturais entre m e n (inclusive). A digitação de pares terminará quando m for maior ou igual a n.

Algoritmo 5: Soma do intervalo

```
leia o número natural m
leia o número natural n
enquanto m < n faça
    soma ← 0                          // acumulador para a soma
    i ← m                             // contador de iterações
    enquanto i < n faça
        soma ← soma + i
        i ← i + 1
    fim enqto
    escreva soma
    leia o número natural m
    leia o número natural n
fim enqto
```

Para saber mais

- Ascencio, A. F. G & Campos, E. A. V. *Fundamentos de programação de computadores: algoritmos, Pascal, C/C++ e Java*. 2. ed. São Paulo: Pearson Prentice Hall, 2007.
- *Doxygen: Main Page*. Disponível em `http://www.stack.nl/~dimitri/doxygen/`
- Marcelo Jo. *Documentação de código - Doxygen*. Disponível em `https://www.embarcados.com.br/documentacao-de-codigo-doxygen/`

Fontes

- Ascencio, A. F. G & Campos, E. A. V. *Fundamentos de programação de computadores: algoritmos, Pascal, C/C++ e Java*. 2. ed. São Paulo: Pearson Prentice Hall, 2007.