

## Problemas com vírgula flutuante

Usando variáveis de tipo `double` (ou `float`), sempre que conveniente, escreva programas em `C` para cada um dos exercícios abaixo.

**P4.1** As funções de variável real  $e^x$ ,  $\sin x$  e  $\cos x$  têm respectivamente, as seguintes expansões em séries de Taylor:

$$\begin{aligned} e^x &= 1 + x + \frac{x^2}{2!} + \dots + \frac{x^n}{n!} + \dots \\ \sin x &= x - \frac{x^3}{3!} + \dots + (-1)^n \frac{x^{2n+1}}{(2n+1)!} + \dots \\ \cos x &= 1 - \frac{x^2}{2!} + \dots + (-1)^n \frac{x^{2n}}{(2n)!} + \dots \end{aligned}$$

- Dado  $x$  e  $n$  calcular as funções usando  $n$  termos das séries. Tenta evitar o cálculo desnecessário de potências de  $x$  e factoriais, escrevendo cada parcela da soma em função da anterior.
- Compara com os valores das funções obtidos utilizando as funções da biblioteca de funções matemáticas do `C`.

**P4.2** A sucessão seguinte converge para  $\pi$ :

$$a_n = 4(1 - 1/3 + 1/5 - \dots + (-1)^{n+1}/(2n-1))$$

Calcular  $a_{1000}$  fazendo sair os resultados para valores de  $n$  de 100 em 100.

**P4.3** Dados os números reais  $a$ ,  $b$  e  $c$  determine as raízes da equação de segundo grau

$$ax^2 + bx + c = 0$$

indicando se as raízes são reais ou complexas.

## Funções

**P4.4** Usando as funções da biblioteca de funções matemáticas do `C`, escreva programas em `C` para:

- Tabelar os valores das funções `seno` e `coseno` para argumentos entre 0 e 90 graus, com intervalos de 10 graus. Nota: As funções aceitam argumentos em radianos.
- Tabelar o valor da função tangente para argumentos entre 0 e 15 graus, por intervalos de 15' (recorde que 1' =  $\frac{1}{60}$  graus).

**P4.5** Para cada uma das funções a baixo escreva um programa em `C` que defina a respectiva função e a table para o intervalo de valores indicado (números reais).

- $f(x) = \cos(\sqrt{x} + 1)$ , para  $x = 0(5)45$ , em graus.
- $f(x) = (-1)^{(1/x)}/(1/x)!$ , para  $x = 0.1(0.05)1.0$ .

**P4.6** Escreva uma função `int verifica(int x, int y, int n)` que retorna 1 se  $x$  e  $y$  estão entre 0 e  $n-1$ , e 0 caso contrário.

**P4.7** Escreva uma função `int multiplo` que tenha como argumentos dois inteiros e que retorne 1 se o segundo é **múltiplo** do primeiro e 0, caso contrário. Testar a função introduzindo uma sequência de pares de inteiros.

**P4.8** Escreva uma função `int digito_decimal(int n, int k)` que retorna o  $k$ -ésimo dígito de um inteiro positivo  $n$ , representado na base 10.