

PM004 - Métodos Numéricos e Aplicações

<http://www.ime.unicamp.br/~campello/pm004>

Atividade 1 - Warm-up: Precisão, Funções, e tudo mais

Data: 28/01/2014

Comandos relevantes caso esteja usando Mathematica:

- `Timing[procedimento]`: Calcula quanto tempo durou a execução de um procedimento. Exemplo: `Timing[1000000!]` nos diz o tempo para calcular 1000000!, seguido do resultado.
- `f[x_]:=` Cria função de x . Exemplo: `f[x_]:=x^2`.
- `If[Condição, O que ocorre, O que ocorre caso contrário]`. Exemplo: `If[x > 1, x^2, 0]` retorna x^2 , caso $x > 1$ e 0 caso contrário.
- `While/For` repetem um comando várias vezes. Exemplo (supondo $x = 10$ de início): `While[x > 0, x = x - 1; Print[x];]` mostra x na tela enquanto $x > 0$.
- Outros comandos: `Table` (cria tabela), `Sum` (somas simples), etc.

Exercício 1. (Calculando o “epsilon da máquina”) Uma maneira de determinar qual a precisão do sistema com o qual estamos lidando é calcular o menor ε tal que $1 + \varepsilon > 1$. Monte um procedimento que calcule computacionalmente ε . Se estiver utilizando Mathematica, compare o resultado com `$MachineEpsilon`.

Exercício 2. (Calculando e). Calcule e , utilizando a Série de Taylor da função e^x , truncada no n -ésimo termo. Quantos termos são necessários para se ter uma aproximação com 5 dígitos corretos? Como justificar isso de maneira teórica?

Exercício 3. (Maneiras de Calcular π) Em geral, métodos clássicos para calcular π utilizam a função arco tangente e sua expansão em Série de Taylor:

$$\arctan x = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \dots$$

Suponha que gostaríamos de calcular π de duas maneiras:

1. Utilizando o fato de que $\pi = 4 \arctan 1$
2. Utilizando a identidade de Euler: $\pi = 20 \arctan \frac{1}{7} + 8 \arctan \frac{3}{79}$.

Monte uma função que calcula π das duas formas. Em cada iteração, utilize apenas uma ordem de aproximação da série de Taylor. Calcule os erros absolutos e relativos em cada iteração. Qual das duas maneiras parece convergir mais rápido? (Difícil) Como justificar isso de maneira teórica?

Exercício 4. (Conjectura de Collatz) Este exercício é um aquecimento para criar funções, utilizar condicionais e loops.

Implemente uma função $f(x)$ que retorne $x/2$, caso x seja par e $3x + 1$, caso x seja ímpar. Itere a sua função n vezes, isto é, a partir de um x_0 da sua escolha, calcule valores x_1, \dots, x_n tais que $x_{i+1} = f(x_i)$. Teste sua função para vários valores de n e vários valores de x_0 . Conjecture o que ocorre à medida que aumentamos n .

Exemplo: $x_0 = 1, n = 5, x_1 = 3 \times 1 + 1 = 4, x_2 = 4/2 = 2, x_3 = 2/2 = 1, x_4 = 3 \times 1 + 1 = 4, x_5 = 2$.

Como você testaria a sua conjectura para vários valores, “automaticamente”?

Exercício 5. Implemente uma função no Mathematica que realize o Método da Bissecção para uma função qualquer. Teste o método para achar uma solução aproximada de $x = \cos[x]$ e $1/x = \ln x$, com pontos iniciais diferentes.

Compare o erro relativo em cada iteração com a fórmula de erro teórica.