CÁLCULO NUMÉRICO - Lista de Exercícios 0
ARITMÉTICA DE PONTO FLUTUANTE

1. Faça uma rotina em MatLab/Octave que soma um milhão (1000000) de vezes o número π (`pi` em MatLab). Depois subtrai do resultado o valor 1000000π. Imprime o resultado.

2. Calcule em MatLab/Octave o maior $n!$ até causar *overflow* (resultado `Inf`) nas precisão simples e dupla. Imprime n e $n!$ (o padrão em MatLab é a precisão dupla, para trabalhar com a precisão simples usem o comando `single(n)`).

3. Analisem as seguintes expressões:

   (a) $\sqrt{x^2+1}-x$

   (b) $\log x - \log y$

   (c) $\sqrt{1+x}-1$

   (d) $(1-\cos x)/(\sin x)$

   para quais valores de x a expressão, calculada pelo MatLab/Octave, pode apresentar um erro relativo significante. Modifique as expressões de forma a minimizar o erro.

4. Considere uma máuina com sistema de representação de números definido por: base 10 ($\beta = 10$), 4 dígitos na mantissa ($t = 5$) e expoente no intervalo: $[-5; 5]$. Pede-se:

   (a) Qual o menor e o maior número em móulo representado nesta máquina?

   (b) Como será representado o número 73758 nesta máuina se for usado o arredondamento? E se for usado o truncamento?

   (c) Se $a = 42450$ e $b = 3$ qual o resultado de $a + b$ se for usado o arredondamento? E se for usado o truncamento? Justifique o resultado.

   (d) Considerando ainda, $a = 42450$ e $b = 3$, qual o resultado da operação $a + \sum_{i=1}^{1} 0b$, considerando que está sendo realizado o truncamento?

   (e) Repetir o item (d), para a operação $\sum_{i=1}^{1} 0b + a$

   (f) Considerando $a = 4245$, $b = 300$ e $c = 100$. Qual o resultado obtido nesta máquina para d e e, calculados de acordo com: $d = (a * b)/c$ e $e = a * (b/c)$. Justifique!

   (g) O que podemos concluir sobre a validade das propriedades como: comutativa, associativa, elemento neutro da adição de núeros em aritmética de ponto flutuante?

5. A precisão da máquina é definida como sendo o menor número positivo em aritmética de ponto flutuante, ε, tal que: $(1 + \varepsilon) > 1$. Este número depende totalmente do sistema de representação da máquina: base numérica, total de dígitos na mantissa, da forma como são realizadas as operaões e do compilador utilizado. É importante conhecermos a *precisão da máquina* porque em vários algoritmos precisamos fornecer como dado de entrada um valor positivo, próximo de zero para ser usado em testes de comparação com zero.

   O algoritmo abaixo estima a precisão da máquina:

Passo 1: $A = 1$
$s = 1 + A$

Passo 2: Enquanto $s > 1$, faça:
$A = A/2$
$s = 1 + A$

Passo 3: Faça $Prec = A * 2$ e imprimir $Prec$

(a) Teste este algoritmo usando o MatLab. Compare `s` obtido com o valor obtido ao se dar o comando `eps` do MatLab. Use o comando `help eps` no MatLab para obter a descriçãao de `eps` no MatLab.

(b) Interprete o passo 3 do algoritmo, isto é, por que a aproximação para $Prec$ é escolhida como sendo o dobro do último valor de A obtido no passo 2?

(c) Na definição de precisão da máquina, usamos como referência o número 1. No algoritmo abaixo a variavel ω é um dado de entrada, escolhido pelo usuário:

Passo 1: $A = \omega$
$s = \omega + A$

Passo 2: Enquanto $s > \omega$, faça:
$A = A/2$
$s = \omega + A$

Passo 3: Faça $Prec = A * 2$ e imprimir $Prec$

(c.1) Teste seu programa atribuindo para ω os numeros : 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}, 10^{-7}, 10^{-8}, 17, 184, 1575, 17893.

(c.2) Para cada valor diferente para ω imprima o valor correspondente obtido para $Prec$. Justifique por que $Prec$ se altera quando ω é modificado.

6. Cálculo de $\exp(x)$: o objetivo é calcular $\exp(x)$ pela série de Taylor até ordem n em torno de *zero*:

$$\exp(x) \approx 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots + \frac{x^n}{n!} \tag{1}$$

(a) Escreva um programa para rodar no MatLab para obter a aproximação para $\exp(x)$ de acordo com a expressão (1). O valor de x e o número de termos da serie, n, são dados de entrada deste programa.

Observe que o cálculo do fatorial: $k!$, necessário na série de Taylor, pode ser feito de modo a evitar o ocorrência de *overflow*. É facil evitar o *overflow*, desde que se observe que o termo (k) pode ser escrito como: $x^k/k! = (x/k)*(x^{k-1}/(k-1)!)$, onde o termo $x^{k-1}/(k-1)!$ já está calculado, pois a série está sendo avaliada a partir do primeiro termo. (Um erro comum no uso da fórmula de Taylor para o cáalculo de $\exp(x)$ é escrever "procedimentos" para avaliar o fatorial: o valor de k é uma dado de entrada e a saída é $k!$. Nestes casos, há ocorrência de *overflow*).

Evitando o *overflow* a série de Taylor pode ser calculada com tantos termos quanto se queira. Qual seria um critério de parada para se interromper o cálculo da série, que não seja a comparação com o valor real de $\exp(x)$?

(b) Teste seu programa com vários valores para x: positivos, negativos, ($x \approx 0$ e x distante de zero) e, para cada valor de x, teste o cálculo da série com vários valores para o número de termos: n. Analise os resultados obtidos.