Universidade Federal de Santa Catarina - UFSC
Centro Tecnológico - CTC
Departamento de Informática e Estatística - INE

Lista de Exercícios 1 - INE5202/INE5232/INE5409

Prof. Sérgio Peters (sergio.peters@ufsc.br)

1) Efetue as seguintes conversões de base:

(a) $(10,1011)_2 = (?)_{10}$

(b) $(10,57)_{10} = (?)_2$

2) Implemente o algoritmo abaixo em um computador com processamento numérico.

```
Início
    inteiro n
    real x
    leia n
    x = 1/n
    imprimir ‘valor inicial x: ’, x (com 30 significativos)
    Para i = 1 até 1
        x = (n + 1) x - 1
        imprimir i , x (com 30 significativos)
    Fim para
Fim
```

Obs.: Teste $n = 2, 3, 10, 16$ e avalie a evolução de x com o número de repetições i. Note que se $x = \frac{1}{n}$ tiver representação exata, $x = (n+1)\cdot\frac{1}{n} - 1 = 1 + \frac{1}{n} - 1 = \frac{1}{n} = x$.

Logo, o valor de x não deveria se alterar com estes cálculos sucessivos, se a primeira definição de x for exata. Explique a razão das diferenças nos valores de x, entre um cálculo e outro, para alguns n.

3) (opcional) Existe uma base onde todo número racional tem representação finita, de acordo com Georg Cantor, “todo racional tem representação finita na base fatorial”.

Conceitualmente, a base fatorial é semelhante à decimal com a diferença de que em um número $X_{F!} = (a_n a_{n-1} \dots a_1, a_{-1} a_{-2} \dots a_{-m})_{F!}$ cada a_i só pode assumir um valor do intervalo $0 < a_i < |i|$, onde a parte inteira é representada por

$$(a_n a_{n-1} \dots a_1,)_{F!} = \left(\sum_{i=n}^{1} a_i i!\right)$$

e a parte fracionária, por

$$(0, a_{-1} a_{-2} \dots a_{-m})_{F!} = \left(\sum_{i=1}^{m} \frac{a_{-i}}{(i+1)!}\right)_{10}.$$

Observe que $(4321)_{F!} = (119)_{10}$ terá como seu sucessor $(10000)_{F!} = (120)_{10}$.

Represente os números na forma fatorada e reconverta para a base decimal:

(a) $(3021)_{F!} = 3\cdot 4! + 0\cdot 3! + 2\cdot 2! + 1\cdot 1! = (77)_{10}$

(b) $(4321)_{F!} = (?)_{10}$

(c) $(10000)_{F!} = (?)_{10}$

(d) $(0,02)_{F!} = \frac{0}{2!} + \frac{2}{3!} = (\frac{2}{6})_{10} = (\frac{1}{3})_{10} = (0,333333\dots)_{10}$

(e) $(0,113)_{F!} = (?)_{10}$
(f) $(321,123)_{F!} = (?)_{10}$

Note que nos exercícios (d), (e) e (f) tem-se representações exatas de números racionais, que na base decimal são dízimas periódicas.

4) Converta os números a seguir para as bases, na ordem indicada:

(a) $(10111,1101)_2 = (?)_{16} = (?)_{10}$
(b) $(BD,0E)_{16} = (?)_2 = (?)_{10}$
(c) $(41,1)_{10} = (?)_2 = (?)_{16}$

Obs.: Indique onde poderá haver perda de dígitos significativos, considerando um número limitado de dígitos representáveis.

5) Na representação $F(2,3,-3,+3)$ com $d_1 \neq 0$ antes da vírgula, não polarizada, calcule:

(a) O número de mantissas representáveis;
(b) O número de expoentes representáveis;
(c) O número de elementos representáveis;
(d) Defina as regiões de *underflow* e *overflow*;
(e) Estime a precisão decimal equivalente;
(f) Proponha uma transformação da representação F apresentada em uma nova com polarização que utilize os limites dos 3 bits totais reservados ao expoente.

6) Na representação $F(2,3,0,7)$ com $d_1 \neq 0$ antes da vírgula e polarização $p = +3$ calcule:

(a) O número de mantissas representáveis;
(b) O número de expoentes representáveis;
(c) O número de elementos representáveis;
(d) Defina as regiões de *underflow* e *overflow*;
(e) Estime a precisão decimal equivalente;

7) Avalie as regiões de *underflow* e *overflow* para a variável de 64 bits, e teste os seus limites em algum compilador numérico (C, Java, ...) ou em um software matemático livre (se usar Octave, pode usar os comandos `single()` e `double()`, para simular variáveis).

8) Avalie a precisão decimal equivalente da variável de 64 bits, através das três formas apresentadas.

9) Simule o algoritmo abaixo:

```
tipo real: e, f, g, h, x, y

h:= 1/2;
x:= 2/3 - h;
y:= 3/5 - h;
e:= (x + x + x) - h;
f:= (y + y + y + y + y) - h;
g:= e/f;
Imprima h,x,y,e,f,g (com 25 dígitos)
```

Obs: Use variáveis reais de 32 e 64 bits, e explique a causa dos resultados obtidos para g, que deveria ser uma indeterminação $\frac{0}{0}$.

10) Para achar as duas raízes de $x^2 + 62,10x + 1 = 0$, utilizando o operador aritmético $F(10,4,-99,+99)$ (quatro dígitos significativos nas operações):

(a) Use a fórmula de Bhaskara normal;
(b) Use a fórmula de Bhaskara normal e racionalizada, sempre com adição das parcelas;
(c) Avalie os erros relativos nas duas formas de avaliação das raízes, sabendo que os seus valores exatos são $x_1 = -0,0161072$ e $x_2 = -62,0839$.

11) Dadas algumas estimativas do valor exato e de valores aproximados numericamente em um algoritmo, avalie o erro absoluto, relativo e percentual, existente nas seguintes situações:

(a) Valor aproximado = 1102,345 e Valor exato = 1100,9.
(b) Valor aproximado = 0,01245 e Valor exato = 0,0119.
(c) Verifique que o erro absoluto obtido, segundo as várias formas de avaliação, pode não refletir a realidade.

12) Dado o número decimal $x = -(10,05)_{10}$:

(a) Calcule os 32 bits da variável IEEE 754 que armazena x;
(b) Calcule o erro de arredondamento percentual exato gerado no armazenamento de x em binário.

13) Dado o número binário $x = (1\ 00000000\ 10000000000000000000001)_2$, armazenado no padrão IEEE de 32 bits (4B). Calcule o decimal x correspondente.

14) Dado o número decimal $y = -(1,51 \cdot 10^{+37})_{10}$:

(a) Calcule os 32 bits da variável y armazenada no padrão IEEE de 32 bits (4B). Mostre explicitamente a parcela binária que foi arredondada;
(b) Calcule o erro exato relativo percentual de arredondamento gerado na conversão de y decimal para binário da variável IEEE de 32 bits.

15) Dado o número decimal exato $y = -(1,1 \cdot 10^{-41})_{10}$:

(a) Calcule os 32 bits da variável y armazenada no padrão IEEE. Mostre explicitamente a parcela binária que foi arredondada;
(b) Calcule o erro exato relativo percentual de arredondamento gerado na conversão de y decimal para binário da variável IEEE de 32 bits.

16) Explique como você calcularia em computador o erro relativo aos arredondamentos acumulados, nos armazenamentos e nas operações usadas, para se obter uma solução numérica qualquer utilizando variáveis de 32 bits. Está disponível também a variável de 64 bits.

17) Implemente um algoritmo que calcule e imprima, o erro de arredondamento da função $exp(x)$ calculada na variável 32 bits, através de aproximação por série de Maclaurin com $n = 4$ em $x = -0,111$, conforme expressão abaixo:

$$exp(x) = e^x \cong 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots + \frac{x^n}{n!}$$

18) Implemente um algoritmo que calcule e imprima, o erro de arredondamento da função $exp(x)$ calculada na variável `double`, através de aproximação por série de Maclaurin com $n = 4$ em $x = -0,111$, conforme expressão abaixo:

$$exp(x) = e^x \cong 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots + \frac{x^n}{n!}$$