Erros de arredondamento

Um número é representado, internamente, na máquina de calcular ou no computador digital através de uma sequência de impulsos elétricos que indicam dois estados: 0 ou 1, ou seja, na base 2 ou binária

De uma maneira geral, um número x é representado na base β por:

$$x = \pm\left[\frac{d_1}{\beta} + \frac{d_2}{\beta^2} + \frac{d_3}{\beta^3} + \ldots + \frac{d_t}{\beta^t}\right].\beta^{\exp}$$

$$0 \le d_i \le (\beta - 1);\ i = 2, 3, \ldots, n.$$

$$\exp_{\min} \le \exp \le \exp_{\max}$$

$$\left[\frac{d_1}{\beta} + \frac{d_2}{\beta^2} + \frac{d_3}{\beta^3} + \ldots + \frac{d_t}{\beta^t}\right]$$ → é chamada de mantissa e é a parte do número que representa seus dígitos significativos e t é o número de dígitos significativos do sistema de representação, **comumente chamado de precisão de máquina**.

Exemplos

No sistema de base β=10, tem-se:

$$0,345_{10} = \left(\frac{3}{10} + \frac{4}{10^2} + \frac{5}{10^3}\right).10^0$$

$$31,415_{10} = 0,31415.10^2 = \left(\frac{3}{10} + \frac{1}{10^2} + \frac{4}{10^3} + \frac{1}{10^4} + \frac{5}{10^5}\right).10^2$$

Os números assim representados estão normalizados, isto é, a mantissa é um valor entre 0 e 1.

No sistema binário tem-se:

$$5_{10} = 101_2 = 0,101 \cdot 2^3 = \left(\frac{1}{2} + \frac{0}{2^2} + \frac{1}{2^3}\right) \cdot 2^3$$

$$4_{10} = 100_2 = 0,1 \cdot 2^3 = \left(\frac{1}{2}\right) \cdot 2^3$$

Exemplos

Numa máquina de calcular cujo sistema de representação utilizado tenha $\beta=2$, $t=10$, $exp_{min}=-15$ e $exp_{max}=15$, o número 25 na base decimal é assim representado:

$$25_{10} = 11001_2 = 0{,}11001 \cdot 2^5 = 0{,}11001 \cdot 2^{101}$$

$$\left(\frac{1}{2}+\frac{1}{2^2}+\frac{0}{2^3}+\frac{0}{2^4}+\frac{1}{2^5}+\frac{0}{2^6}+\frac{0}{2^7}+\frac{0}{2^8}+\frac{0}{2^9}+\frac{0}{2^{10}}\right)\cdot 2^{101}$$

Ou de forma mais compacta:

1 1 0 0 1 0 0 0 0 0	1 0 1
Mantissa	**Expoente**

Cada dígito é chamado de bit, portanto, nesta máquina são utilizados 10 bits para a mantissa, 4 bits para o expoente e mais um bit para o sinal da mantissa (se bit=0 positivo, se bit=1 negativo) e um bit para o sinal do expoente, resultando no total de 16 bits, que são assim representados:

0	1 1 0 0 1 0 0 0 0 0	0	0 1 0 1
Sinal da mantissa	**Valor da Mantissa**	**Sinal do expoente**	**Valor do Exponte**

Exemplos

Utilizando a mesma máquina do exemplo anterior, $3{,}5_{10}$ seria dada por:

$$3{,}5_{10} = 0{,}111 \cdot 2^{10}$$

0	1 1 1 0 0 0 0 0 0 0	0	0 0 1 0

Ainda utilizando a mesma máquina do exemplo anterior, o número $-7{,}125_{10}$ seria assim representado dada por:

$$-7{,}125_{10} = -0{,}111001 \cdot 2^{11}$$

1	1 1 1 0 0 1 0 0 0 0	0	0 0 1 1

O maior valor representado por esta máquina seria:

0	1 1 1 1 1 1 1 1 1 1	0	1 1 1 1

$$0{,}1111111111 \cdot 2^{1111} = 32736_{10}$$

O menor valor seria:

$$-0{,}1111111111 \cdot 2^{1111} = -32736_{10}$$

Logo, os números que podem ser representados nesta máquina estariam contidos no intervalo [-32736 ; 32736]

Exemplos

Nesta máquina, ainda, o valor zero seria representado por:

0	0 0 0 0 0 0 0 0 0 0	0	0 0 0 0

O próximo número positivo representado seria:

0	1 0 0 0 0 0 0 0 0 0	1	1 1 1 1

$$0,1 \cdot 2^{-15} = 0,000015259$$

O subsequente seria:

0	1 0 0 0 0 0 0 0 0 1	1	1 1 1 1

$$0,1000000001 \cdot 2^{-15} = 0,000015289$$

Ao se tentar representar números reais por meio deste sistema, certamente se incorre nos chamados erros de arredondamento, pois nem todos o números reais tem representação no sistema.

Exemplos

$$0,1_{10} = 0,0001100110011\ldots_2$$

O valor decimal 0,1 tem como representação binária um número com infinitos dígitos, logo, ao se representar 0,1 nesta máquina comete-se um erro, pois:

0	1 1 0 0 1 1 0 0 1 1	0	0 0 1 1

$$= 0,099976_{10}$$

Uma fração racional na base 10 pode ser escrita, exatamente com um número finito de digitos binários somente se puder ser escrita como o quociente de dois inteiros r/s, onde $s=2^N$ para um inteiro N. Infelizmente, apenas uma pequena parte das frações racionais satisfaz esta condição.

Exemplos

Como ilustração, apresenta-se abaixo os sistemas de representação de algumas máquinas:

Máquinas	β	t	exp_{min}	exp_{max}
Hewlett-Packard 45	10	10	-98	100
Texas SR-5X	10	12	-98	100
PDP -11	2	24	-128	127
IBM/360	16	6	-64	63
IBM/370	16	14	-64	63

Exemplos

Um parâmetro que é muito utilizado para se avaliar a precisão de um determinado sistema de representação é o número de casas decimais exatas da mantissa e este valor é dado pelo valor decimal do último bit da mantissa, ou seja, o bit de maior significância. Logo:

$$\text{Precisão} \leq \frac{1}{\beta^t}$$

Exemplos

Numa máquina com $\beta = 2$ e $t = 10$, a precisão da mantisa é da ordem de

$$\frac{1}{2^{10}} = 10^{-3}$$

Logo o número de dígitos significativos da mantissa é 3.

Ressalta-se a importância de se saber o número de dígitos significativos do sistema de representação da máquina que está sendo utilizada para que se tenha noção da precisão do resultado.

Exemplos

Consideremos um equipamento com o SPF normalizado (b, n, expmin, expmax)= SPF (10,4,-5,5)

a) Se a= 0.5324×10^{3} e b=0.4212×10^{-2} , então a x b = 0.22424688×10^{1}

Que é arredondado e armazenado como $(a \times b)_a = 0.2242 \times 10^{1}$

b) Se a= 0.5324×10^{3} e b=0.1237×10^{2} , então a + b = 0.54477×10^{3}

Que é arredondado e armazenado como $(a + b)_a = 0.5448 \times 10^{3}$

Erro Absoluto

Definimos erro absoluto como:

$$E_{abs} = \left| a_{\mathrm{ex}} - a_{aprox} \right|$$

Onde a_{ex} : é o valor exato da grandeza considerada e a_{aprox} é o valor aproximado da mesma grandeza.

Como na maioria das vezes o valor exato não é disponível, a definição anterior fica sem sentido. Assim, é necessário trabalhar-se com um limitante superior para o erro, isto é, escrevê-lo na forma:

$$\left| a_{ex} - a_{aprox} \right| \leq \varepsilon$$

Onde ε é um limitante conhecido.

Erro Absoluto

A desigualdade anterior pode ser entendida da seguinte maneira:

$$-\varepsilon \leq \left(a_{ex} - a_{aprox}\right) \leq \varepsilon$$

Ou ainda:

$$\left(a_{aprox} - \varepsilon\right) \leq a_{ex} \leq \left(a_{aprox} + \varepsilon\right)$$

Isto é, a_{aprox} é o valor aproximado da grandeza a_{ex} com erro absoluto não superior a ε.

Erro Relativo

Definimos erro relativo como:

$$E_{rel} = \left| \frac{E_{abs}}{a_{aprox}} \right| = \frac{\left| a_{ex} - a_{aprox} \right|}{\left| a_{aprox} \right|}$$

Onde a_{ex} : é o valor exato da grandeza considerada e a_{aprox} é o valor aproximado da mesma grandeza. Como na maioria das vezes o valor exato não é disponível, a definição anterior fica sem sentido. Assim, é necessário trabalhar-se com um limitante superior para o erro relativo, isto é, escrevê-lo na forma:

$$\delta \leq \left| \frac{\varepsilon}{a_{aprox}} \right|$$

Onde δ é um limitante conhecido.

Pode-se observar que o erro relativo nos fornece mais informações sobre a qualidade do erro que estamos cometendo num determinado cálculo, uma vez que no erro absoluto não é levada em consideração a ordem de grandeza calculada

Erro Relativo

Exemplo:

a) Consideremos o valor exato a_{ex} = 2345.713 e o valor aproximado a_{aprox} = 2345.000.

Então,

E_{abs} = 0.713

E_{rel} = 0.00030396

b) Consideremos o valor exato a_{ex} = 1.713 e o valor aproximado a_{aprox} = 1.000

Então,

E_{abs} = 0.713

E_{rel} = 0.416229

Exercícios

1) Considere o sistema SPF (3,3,-2,1):

a) Quantos números podemos representar neste sistema?
b) Represente no sistema os números : $x_1 = (0.40)_{10}$ e $x_2 = (2.8)_{10}$

2) Considere o sistema SPF (2,5,-3,1):

a) Quantos números podemos representar neste sistema?
b) Qual o maior número na base 10 que podemos representar neste sistema?

3) Calcule e represente na reta os positivos representáveis do sistema de ponto flutuante normalizado SPF (3, 2, -1, 2).

4) No sistema de ponto flutuante normalizado SPF (2,3,-1, 2) represente em cada caso, o valor arrendondado das seguintes expressões:

a) $0.101 \times 2^0 + 0.110 \times 2^{-1}$
b) $0.101 \times 2^0 + 0.111 \times 2^1$
c) $0.111 \times 2^0 + 0.110 \times 2^{-1}$