Escritora Margarida Fonseca Santos apresenta livro na Ribeira Grande

Foto:CMRG

A biblioteca municipal Daniel de Sá, na Ribeira Grande, recebe esta noite. a partir das 19h00, a escritora Margarida Fonseca Santos que vai apresentar o seu mais recente trabalho: Altamente – Livro de Treino Mental.

Margarida Fonseca Santos é uma escritora de reconhecido talento que conta com mais de uma centena de obras publicadas. Escreve ficção e teatro para adultos e crianças, mas a maioria dos seus livros inserem-se na área infanto-juvenil, com muitas das suas obras referenciadas no Plano Nacional de Leitura.

Nesta Quinta-feira estará na Ribeira Grande para apresentar e falar do seu trabalho, Altamente – Livro de Treino Mental, bem como de outros assuntos que os participantes queiram abordar. A participação é gratuita.

A sinopse do livro levanta a ponta do véu do que se poderá encontrar no mesmo. "Todos nós, num momento ou noutro, já tivemos a sensação de lutar com a nossa mente, sem saber como dominá-la. Quando as coisas correm bem, achamos que tivemos sorte. Quando correm mal, achamos que somos incapazes, fracos ou que alguém nos boicotou. Será verdade? Não, de todo. Quase sempre, fomos nós, a nossa mente, a provocar o desfecho, bom ou mau. É sobre isso que vamos falar! Vamos descobrir como podemos ser mais positivos, mais atentos e eficazes, mais seguros".

Aula de Zumba nas piscinas da Lagoa

A Câmara Municipal de Lagoa vai isentar a entrada no Complexo de Piscinas Municipal no próximo domingo, dia 11 de Setembro, último dia da época balnear oficial no concelho.

O ponto alto deste último dia será a realização de uma aula de zumba para que possa despedir-se do verão com energia e boa disposição. Recorde-se que, este complexo de piscinas ostenta pela 20ª vez consecutiva as cores do galardão máximo de excelência balnear, a Bandeira Azul. A Bandeira Azul é um prémio atribuído anualmente por um júri internacional a praias que cumpram um conjunto de critérios de natureza ambiental, de segurança e conforto dos utentes, informação e sensibilização ambiental. O Complexo Municipal de Piscinas é uma estrutura balnear de excelência e a presente época balnear foi efectivamente de mais uma época de sucesso. Aliás, o facto deste complexo municipal ostentar ininterruptamente a Bandeira Azul.

A codificação binária da informação - formatos para os números fracionários

Por: Jerónimo Nunes
Docente da Universidade dos Açores
jeronimo.am.nunes@uac.pt

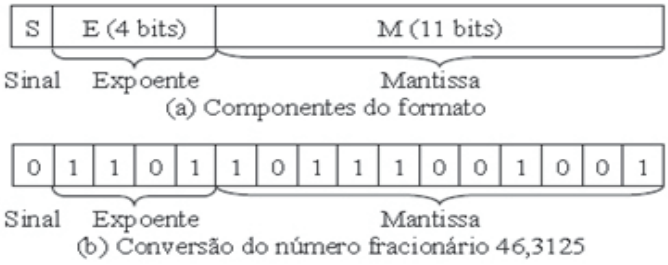

Figura 1 - Formato de vírgula flutuante de 16 bits

Na sequência do artigo "A codificação binária da informação - códigos numéricos bipolares", publicado neste jornal no passado dia 4 de agosto, primeiro de uma série dedicada à temática da Codificação da Informação, onde foram abordados os métodos usados na representação binária dos números inteiros positivos e negativos, neste artigo serão apresentados os métodos específicos para a codificação dos números com parte fracionária.

Os números fracionários, os que são representados por uma dízima finita ou por uma dízima infinita periódica, ou classicamente em forma de fração irredutível, são escritos seguindo as mesmas regras usadas para os números inteiros, definidas no sistema de numeração: recorre-se ao mesmo conjunto de algarismos e à mesma uma notação posicional. Os pesos associados a cada algarismo da parte fracionária são potências da base do sistema de numeração, mas com expoente negativo, isto é, frações inversas de potências da base.

No sistema decimal, o número fracionário 0,375 pode escrever-se como 3x(0,1)+7x(0,01)+5x(0,001), ou, utilizando a potenciação, 3x(0,1)^1+7x(0,1)^2+5x(0,1)^3, ou, evidenciando o inverso da base do sistema de numeração 3x(1/10)^1+7x(1/10)^2+5x(1/10)^3 (0,1=1/10).

Analogamente, no sistema binário, o número 0,011 será escrito numa forma não condensada como 0x(1/2)^1+1x(1/2)^2+1x(1/2)^3, que em base decimal corresponderá a 0x1/2+1x1/4+1x1/8 (=0,375). Este processo de conversão da representação binária para decimal de uma fração segue diretamente as normas dos sistemas de numeração. Para o processo inverso, a conversão decimal-binário, um dos métodos possíveis consiste em multiplicar a fração decimal por 2 e considerar o valor da parte inteira resultante como o bit mais à esquerda (de maior peso) da fração binária; para obter os restantes bits, sucessivamente de menor peso, repete-se processo até se obter uma fração decimal igual a zero. Por exemplo, 0,375x2=0,750; 0,750x2=1,500; 0,500x2=1,000; logo 0,375 convertida para binário será 0,011. Enquanto na conversão decimal binário de números inteiros há a garantia do processo ser finito (ao fim de um certo número de divisões por 2 do decimal inteiro resultará um quociente igual a zero), na conversão de uma fração a multiplicação por 2 poderá nunca resultar num número com parte fracionária igual a zero produzindo uma sequência infinita de bits. Em termos práticos, define-se um número máximo de bits para a parte fracionária o que poderá originar um erro na conversão. Por exemplo, considerando um máximo de 7 bits, a conversão da fração decimal 0,2 para binário seria efetuada deste modo: 0,2x2=0,4; 0,4x2=0,8; 0,8x2=1,6; 0,6x2=1,2; 0,2x2=0,4; 0,4x2=0,8; 0,8x2=1,6; obtendo-se 0,001101. Convertendo esta fração binária para decimal obteríamos 1/8+1/16+/128 (=0,1953125) que difere de 0,2 em 0,0046875, o erro cometido na codificação. Este erro será menor quanto maior for o número de bits que reservarmos para a parte fracionária.

Os códigos BCD podem também ser usados para codificar números com parte fracionária, apresentando a vantagem de não originarem erros da conversão porque cada algarismo decimal é codificado separadamente no correspondente binário.

Além da codificação da parte inteira e da parte fracionária dos números, recorrendo a métodos distintos, e da codificação do sinal usando um bit ("0" para o sinal "+" e "1" para o sinal "-") é necessário uma indicação quanto à posição da vírgula para que um computador possa efetuar operações com este tipo de números. O conjunto de regras usado na representação dos números designa-se por formato dos números e dele fazem parte: (1) a notação utilizada na escrita, (2) a escolha das codificações adequadas para as componentes inteira e fracionária e para o expoente e (3) o número de bits a usar para codificação de cada componente. No chamado formato de vírgula flutuante, os números são representados usando quatro componentes S (sinal), M (mantissa ou parte significativa), B (base) e E (expoente, negativo ou positivo) com a seguinte notação: SMxB^E, também conhecida por notação científica. O número 1230000 pode ser escrito nesta notação como +1230000x10^0; ou 123,0000x10^4; ou 0,123000x10^7; ou por outras expressões matematicamente equivalentes. Na primeira e última representação, ditas normalizadas, não é necessária a indicação da posição da vírgula porque está subentendida, daqui resultando mantissas apenas, respetivamente, inteiras e fracionárias. O número de bits destinado à codificação da mantissa (M) determina a precisão dos números representados e o número de bits reservados ao expoente (E) determina o maior valor numérico que é possível representar, no caso de expoentes positivos, ou o valor mais próximo de zero, no caso de expoentes negativos.

Quando o número de bits da mantissa for insuficiente, será necessário efetuar um truncatura ou um arredondamento que provocará um erro de precisão na representação do número fracionário. Se o valor do número a representar for muito elevado o número de bits do expoente poderá não ser suficiente ocorrendo um "overflow" (transbordo) em que é ultrapassada a capacidade do formato para codificar o número. Quando não for possível representar um número muito pequeno (próximo de zero) por insuficiente precisão do formato de vírgula flutuante diz-se que ocorreu um "underflow".

Na codificação da mantissa fracionária (com sinal) recorre-se ao código sinal e valor absoluto e do expoente (positivo ou negativo) recorre-se ao código binário deslocado. Este último é um tipo particular de código usado para números com sinal que são codificados adicionando uma constante ao número para se obter um valor não negativo que, por sua vez, será convertido para binário pelo método usual.

A conversão de um número decimal fracionário para formato de vírgula flutuante efetua-se em dois passos: primeiro converte-se separadamente para binário a parte inteira e a parte fracionária, usando os métodos apropriados, e depois escreve-se o binário fracionário resultante de acordo com as codificações e cumprimentos definidos no formato de vírgula flutuante.

Por exemplo, a conversão do número 46,3125 para um formato de vírgula flutuante de 16 bits, com 11 bits para a mantissa (e 1 para o sinal) e 4 bits para o expoente, efetua-se do seguinte modo (ver a figura 1): parte inteira 101110; parte fracionária 01001; número escrito na notação normalizada 0,10111001001x2^6; ou seja, S=1, M=10111001001. Para codificar o expoente adiciona-se 7 e converte-se para binário: E=6+7=13, em binário 1101.

Para uniformização da representação computacional dos números nos microprocessadores dos vários fabricantes foi estabelecido o standard internacional IEEE 754 que define dois possíveis formatos de vírgula flutuante: precisão simples, com o cumprimento de 32 bits, e precisão dupla, com o comprimento de 64 bits. As caraterísticas mais relevantes destes formatos são apresentadas na figura 2. Estes formatos reservam combinações especiais de valores para representar abstrações matemáticas que não podem ser escritas usando algarismos (como o infinito) e resultados de operações impossíveis, como as conhecidas indeterminações ou os valores de funções matemáticas fora do seu domínio (raiz quadrada ou logaritmo de valores negativos). A existência de uma combinação especial que represente o "infinito" tem utilidade no cálculo de valores de funções trigonométricas que possuem valores finitos para aquele argumento.

Utilizando igual número de bits, quer num formato apenas para números inteiros quer num formato de vírgula flutuante, são representados o mesmo número de valores, mas o segundo formato permitirá representar números de maior magnitude. Num determinado intervalo, todos os números inteiros são computacionalmente representáveis enquanto apenas alguns dos números fracionários são representáveis num formato de vírgula flutuante porque qualquer intervalo contém uma quantidade infinita de fracionários.

As linguagens de programação consideram dois tipos de valores numéricos (dados): inteiros "int" (integer) e fracionários "float", de precisão simples, e "double", de precisão dupla. Com o tipo "int" as operações aritméticas são menos complexas e mais rápidas, mas com o tipo "float" ou "double" as operações aritméticas requerem maior esforço computacional. A complexidade das operações com formatos de vírgula flutuante resulta da manipulação de vários componentes numéricos com diferentes codificações e da exigência de lidar com valores especiais e casos excecionais. Os primeiros microprocessadores não possuíam hardware que executasse diretamente operações com números fracionários - eram efetuadas executando software dedicado. Posteriormente, os microprocessadores passaram a incluir uma unidade aritmética dedicada a este tipo de operações denominada "floating-point unit" (FPU).

Caraterística	Formato de 32 bits	Formato de 64 bits
Comprimento: nº de bits	32	64
Expoente: base; n.º bits	2; 8	2; 11
Expoentes: mínimo; máximo	-126; 127	-1022; 1023
Número de expoentes	254	2046
Mantissa: n.º bits; n.º de valores	23; 2^{23}	52; 2^{52}
Gama de valores	10^{-38}, 10^{38}	10^{-308}, 10^{308}
Número de valores	$1{,}98 \times 2^{31}$	$1{,}99 \times 2^{63}$

(a) Caraterísticas gerais dos formatos

Sinal	Expoente	Mantissa	Tipo de número
0/1	0	0	Zero
0/1	111...111	0	± Infinito (∞)
0/1	111...111	≠ 0	Não válido (Not a Number)

(b) Combinações especiais de valores

Operação	Resultado	Operação	Resultado
$+\infty \times +\infty$	$+\infty$	$+\infty - +\infty$	NaN
$-\infty - -\infty$	$-\infty$	$0 \div 0$	NaN
± (não zero) ÷ 0	$\pm\infty$	$+\infty \div +\infty$	NaN
(não zero) ÷ ±∞	0	$\pm\infty \times 0$	NaN

(c) Resultados especiais de operações aritméticas

Figura 2 - Standard IEEE 754 para formatos de vírgula flutuante