Diocese de Angra em destaque no Semanário Ecclesia entre 20 destinos a visitar este Verão

Contemplar a natureza, conhecer o património religioso e participar nas festas em diferentes regiões do país são propostas do semanário Ecclesia, uma em cada diocese, para tempo de férias.

A última edição de verão apresenta entre os 20 destinos a diocese de Angra, nomeadamente a ilha das Flores, num texto assinado pelo diretor do Serviço Diocesano da Pastoral das Comunicações Sociais da Igreja, Cónego Ricardo Henriques e que pode ler aqui no Sítio Igreja Açores.

"20 destinos para conhecer Portugal… Uns mais improváveis que outros, mas todos seguramente merecedores de uma visita. Num roteiro por todas as dioceses portuguesas, escolhemos a natureza, o património, os templos e sobretudo as pessoas para lhe propor 'ir para fora cá dentro…'", refere o texto de apresentação desta edição.

O texto intitulado "Ilha das Flores, uma marca de Deus" apresenta vários lugares sublimes onde a natureza convida à vivência da fé.

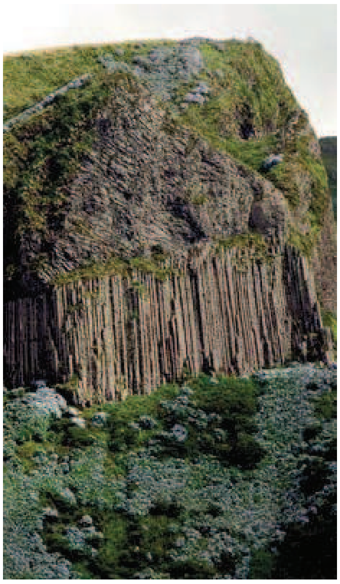
"Silêncio, durante o dia, aliado à sinfonia dos pássaros com raiar da madrugada, é um cenário natural ideal para o encontro com Deus" refere o sacerdote.

No editorial desta edição, Paulo Rocha refere-se ao "humanismo" que marca qualquer visita, a que se faz "a quem está só" como a "finalmente conseguida" a amigos ou sítios de referência,

"Na visita que o turismo sugere, é urgente descentralizar destinos, ter em consideração cada vez mais as periferias e estar permanentemente em saída. Porque o belo não habita apenas a praça conhecida, o templo de celebrações solenes ou os caminhos de rotinas diárias", refere.

"A paz, o conhecimento e a boa proximidade entre pessoas e comunidades acontece muitas vezes depois de uma visita! Vamos a isso!", conclui o editorial do semanário Ecclesia, uma edição digital que é retomada no dia 2 de Setembro.

Foto: IA



A codificação binária da informação - códigos numéricos bipolares

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

Os computadores são constituídos por circuitos digitais que armazenam e processam informação binária, escrita ou representada usando dois símbolos, os bits, "0" e "1", resultando em sequências de comprimento variável, de acordo com o volume da informação a representar. A informação com que lidamos no dia-a-dia não se encontra escrita em linguagem binária: para escrevermos os números usamos o sistema de numeração decimal e para as palavras usamos os caracteres alfabéticos, os sinais de pontuação e outros símbolos. Para traduzir a informação não binária para informação binária é necessário fazer corresponder aos números decimais e aos caracteres da linguagem escrita os respetivos equivalentes binários. Este processo é conhecido por codificação e as regras que o regem - o código - são distintas consoante o tipo de informação (numérica ou alfabética) a que se aplicam. Um código define também as regras que permitem reverter a codificação, ou seja, a partir da informação binária obter a informação não binária, processo denominado descodificação.

Um sistema de numeração define as normas de escrita de números de qualquer grandeza usando um conjunto reduzido de símbolos (os algarismos), em quantidade igual à base do sistema, e uma notação que associa a cada algarismo um peso determinado pela sua posição relativa no número. No sistema decimal, aos algarismos do número 123 estão associados, respetivamente, os pesos 100, 10 e 1, ou seja, pode escrever-se 1x100+2x10+3x1 ou, evidenciando a base do sistema (=10), 1x10^2+2x10^1+3x10^0 (^ é usado como símbolo da operação de potenciação). Analogamente, no sistema binário, o número 1100 escrito de uma forma não condensada será 1x2^3+1x2^2+0x2^1+0x2^0, que em decimal corresponde a 1x8+1x4+0x2+0x1 (=12). Este processo de conversão da representação binária para decimal de um número inteiro segue diretamente as normas dos sistemas de numeração. Para o processo inverso, a conversão decimal binário, um dos métodos possíveis consiste em dividir o decimal por 2 e sucessivamente os quocientes resultantes, até se obter um quociente 0, e escrever os restos pela ordem inversa da sua obtenção. Por exemplo, 12:2=6+0; 6:2=3+0; 3:2=1+1; 1:2=0+1; logo, 12 convertido em binário resulta 1100.

Na codificação binária, os números positivos e negativos são distinguidos acrescentando, à esquerda dos bits que codificam o valor, um outro que representa o sinal: "0" para o sinal "+" e "1" para o sinal "-". A representação dos números com sinal obtém-se aplicando os chamados códigos bipolares, alguns dos quais se apresentam na figura 1. Um código bipolar deverá ser definido tendo em consideração a facilidade de cálculo do valor codificado e de execução das operações aritméticas. No código sinal e valor absoluto, os números são codificados convertendo o valor do número (sem o sinal) para binário e acrescentado o bit de sinal à esquerda. Com este processo obtêm-se duas representações para o número zero - "+0" e "-0" e, usando o mesmo método da adição decimal, nem sempre se verifica a propriedade dos valores simétricos. Por exemplo, 101(-1) + 001(+1) = 110(-2), quando deveria ser zero.

Nos outros códigos bipolares, recorre-se ao conceito complemento de Base-1 (=2-1) de um número N, com n dígitos, definido pela expressão C1=2^n-N-1. Considerando números com 3 bits, o complemento de 1 de N é o valor que adicionado a N tem como resultado o maior número que é possível representar com 3 bits - 111 (=7). O interesse prático deste código advém da facilidade em obter o valor codificado de um número negativo sem recorrer à operação de subtração: depois de converter o valor do número (sem o sinal) para binário e acrescentar o bit "0" à esquerda, trocam-se os valores dos bits (de 0 para 1 e de 1 para 0) para obter o respetivo complemento. O código complemento de 1 apresenta duas codificações para o valor zero, como o código sinal e valor absoluto, mas já verifica a propriedade da adição dos simétricos que, para números com 3 bits, será sempre igual a 111, uma das representações do valor zero.

Para superar esta ambiguidade na representação do zero, recorre-se ao conceito de complemento de Base (=2) de um número N, com n dígitos, definido pela expressão C2=2^n-N, relacionado com o complemento de Base-1 pela expressão C2=1+C1. Esta igualdade permite um processo expedito de codificação dos valores negativos a partir da codificação prévia em complemento de 1. Da aplicação do código complemento de 2 resulta uma representação única para o zero e a codificação adicional de um número negativo - é um código assimétrico. O valor 100 corresponde a -4 em decimal porque da adição com 011(3) resulta 111(-1). Aplicando as regras de codificação dos códigos de complemento a um número negativo obtemos o simétrico em binário, um número positivo, que depois será convertido para se conhecer o corresponde valor decimal.

Com o código complemento de 2, a subtração X-Y é transformada na adição X+(-Y), em que o simétrico de Y se obtém aplicando o código complemento de 2, possibilitando efetuar estas duas operações aplicando o mesmo método de cálculo, contrariamente ao que acontece na aritmética decimal.

Poderemos efetuar a codificação dos números sem recorrer às propriedades dos sistemas de numeração convertendo separadamente cada algarismo decimal no correspondente valor binário. Neste tipo de códigos, designados BCD (Binary-Coded Decimal), mostrados na figura 2, são necessários 4 bits para codificar os 10 algarismos embora não sejam utilizadas todas as combinações binárias disponíveis (=16). No primeiro tipo destes códigos, o BCD 8-4-2-1, cada número de 0 a 9 é convertido para binário seguindo o processo anteriormente referido. A adição de dois números em BCD efetua-se aplicando o método usual, mas tendo em atenção que o resultado poderá ser um valor binário que não corresponde a um algarismo decimal, havendo necessidade de um ajustamento. Por exemplo, 0101(5)+0111(7)=1100, valor que não tem correspondência em BCD e a que deverá ser adicionado 0110(=16-10) para se obter o resultado correto em BCD 1100+0110=1 0010, que em decimal será 12. A subtração em BCD recorre aos conceitos complemento de Base-1 (=9) e complemento de Base (=10) para obter o simétrico de um número: o complemento de 9 de um número decimal, com um algarismo, é o valor que adicionado ao próprio número é igual a 9; o complemento de 10 obtém-se por adição de 1 a este valor. Para efetuar a subtração 7-5 adicionamos os valores BCD correspondentes a 7 e a 4 (que é o complemento de 9 do número 5). Assim, teremos 0111+0100=1011, a que adicionamos 1 (para cálculo do complemento de 10) e ajustamos para obter o resultado correto 1100+0110=0010(2), ignorando o último bit de transporte.

No código BCD 8-4-2-1 o complemento de 9 (em decimal) não corresponde ao complemento de 1 (em binário): o complemento do decimal 5 é 4, mas o complemento de 0101 é 1010, uma combinação inválida neste código BCD. No código BCD 2-4-2-1 o peso do bit mais à esquerda (ou mais significativo) é 2 e não 8, resultando numa codificação distinta dos decimais, em que os números de 2 a 7 possuem duas codificações alternativas. Escolhendo a codificação mais esquerda no quadro da figura 2, este código apresentará a propriedade de o valor BCD correspondente ao complemento de 9 do decimal ser obtido aplicando o complemento de 1 ao binário, isto é, o BCD 2-4-2-1 do decimal 5 é 1011 e o respetivo complemento de 9 é 4, que corresponde a 0100 em BCD, o complemento de 1 de 1011. Neste código, dito auto complementar, para efetuarmos a subtração 7-5 convertemos estes valores para BCD obtendo, respetivamente, 1101 e 1011 e adicionamos 1 ao complemento de 1 do segundo valor 0100+1=0101. Este valor adicionado ao BCD de 7, 1101+0101=0010(2), produzirá o resultado correto da subtração inicial.

A título de exemplo de um outro tipo de código BCD, apresenta-se na figura 2 o código excesso 3 que codifica os dígitos decimais em binário natural após adicionar três unidades ao valor de cada dígito. Este é também um código auto complementar.

Nos computadores atuais, a representação dos números inteiros, positivos e negativos, faz-se recorrendo ao código complemento de 2, usando um número fixo de bits, geralmente 32 ou 64. Nos números positivos, os bits à esquerda não necessários para a codificação do valor numérico serão iguais a 0, mas nos números negativos serão iguais a 1. O código complemento de 2 permitir a aplicação do mesmo método para realizar as operações de adição e de subtracção que é análogo ao utilizado na aritmética decimal. No que respeita à multiplicação com valores codificados no código complemento de 2, já não será possível usar o método análogo ao da aritmética decimal. No sistema decimal, a operação -2x(-1), expressa como 111x110, em complemento de 2, terá como resultado 101010, a corresponde -22, em decimal.

Foto: DR

Equivalente decimal	Sinal e valor absoluto	Complemento de 1	Complemento de 2
+3	011	011	011
+2	010	010	010
+1	001	001	001
+0	000	000	000
-0	100	111	000
-1	101	110	111
-2	110	101	110
-3	111	100	101
-4	-	-	100

Figura 1 - Códigos bipolares

Dígitos decimais	Código BCD (8-4-2-1)	Código BCD (2-4-2-1)	Código Excesso 3
0	0000	0000	0011
1	0001	0001	0100
2	0010	0010 ou 1000	0101
3	0011	0011 ou 1001	0110
4	0100	0100 ou 1010	0111
5	0101	1011 ou 0101	1000
6	0110	1100 ou 0110	1001
7	0111	1101 ou 0111	1010
8	1000	1110	1011
9	1001	1111	1100

Figura 2 - Códigos BCD