



Campus de Gualtar
4710-057 Braga



UNIVERSIDADE DO MINHO
ESCOLA DE ENGENHARIA

Departamento de
Informática

Conceitos de Sistemas Informáticos

Módulo: Arquitectura de Computadores

Exercícios: Representação de dados

Alberto José Proença

05-Mar-02

1. Efectue as seguintes conversões de base:
 - a) De 10110101101010_2 para hexadecimal
 - b) De $0xC407$ para octal
 - c) De 230.12_4 para decimal
 - d) De 725_8 para ternário
2. Mostre, em binário, o conteúdo de um registo de 16 bits com os seguintes números, nas várias representações binárias possíveis (sinal+magnitude, complemento para 1, complemento para 2, e representação por excesso 2^{n-1}):
 - a) -120
 - b) -25
 - c) -58
 - d) 33
3. Considere a seguinte sequência de bits armazenada num registo do Pentium:
 $1111\ 1110\ 1111\ 1111\ 1111\ 1111\ 0000\ 0000$
Indique, em decimal, o valor que ela representa, assumindo que o registo contém:
 - a) Um valor da classe dos inteiros
 - b) Um apontador para a memória
 - c) Um valor da classe dos reais
4. A dimensão máxima de RAM que os primeiros PC's podiam ter, em *bytes*, era de 640K. Considere que este valor se encontra armazenado na memória dum Pentium a partir da localização $0x800040C0$. Mostre, em hexadecimal, o conteúdo das células de memória (e respectivo endereço) que contém esse valor, considerando que esse valor foi armazenado como:
 - a) Texto
 - b) Inteiro (com sinal)
 - c) Real (precisão simples)

Para os próximos exercícios considere 2 novos formatos de vírgula flutuante baseados na norma IEEE:

- formato `LITTLE`:
 - 8 bits
 - o bit mais significativo contém o bit do sinal
 - os 4 bits seguintes formam o expoente (em excesso de 7)
 - os últimos 3 bits representam a mantissa
- formato `TINY`:
 - 6 bits
 - o bit mais significativo contém o bit do sinal
 - os 3 bits seguintes formam o expoente (em excesso de 3)
 - os últimos 2 bits representam a mantissa

Para todos os restantes casos, as regras são as mesmas que as da norma IEEE (valor normalizado, desnormalizado, representação do 0, infinito, e NaN).

5. Para ambos os formatos, apresente os seguintes valores em decimal:
 - a) O maior número finito positivo
 - b) O número positivo normalizado mais próximo de zero
 - c) O maior número positivo desnormalizado
 - d) O número positivo desnormalizado mais próximo de zero
6. Codifique os seguintes valores como números de vírgula flutuante no formato `LITTLE`: $3/4$, $-13/16$, 44 e -104
7. Calcule os valores (n.º real, ± 8 , NaN) correspondentes aos seguintes padrões de bits no formato `LITTLE`:
 - a) 10110011
 - b) 01111010
 - c) 10010001
 - d) 01001111
 - e) 11000001
8. Converta os seguintes números `LITTLE` em números `TINY`. *Overflow* deve ser representado por ± 8 , *underflow* por ± 0 e arredondamentos deverão ser para o valor par mais próximo.
 - a) 00010000
 - b) 11101001
 - c) 00110011
 - d) 11001110
 - e) 11000101