UFSC / CTC / INE
Disciplina: Introdução à Informática

Curso de Sistemas de Informação: INE5602-138

Prof. Dr. João Dovicchi*

Solução dos Exercícios em sala 2

1. Converta os números decimais para números binários, com bit de sinal:

a. $+142$

Para representar o 142 com bit de sinal precisamos de 9 bits, pois ele está entre -256 e +255 (512 valores). Resposta: `010001110`

b. -231

Para o número -231 também precisamos de 9 bits. Como ele é negativo, temos que encontrar o número +231 com 9 bits e tirar o complemento de 2 para encontrá-lo na sua forma negativa (com o bit de sinal). Resposta:

$$
\begin{array}{rcr}
+231_{10} & \rightarrow & \texttt{011100111} \\
\text{complemento} & \rightarrow & \texttt{100011000} \\
\text{compl. de 2 (soma 1)} & \rightarrow & \texttt{+1} \\
\cline{3-3}
-231_{10} & \rightarrow & \texttt{100011001}
\end{array}
$$

c. -1

O -1 pode ser representado por, no mínimo, 2 bits (entre -2 e +1) ou com qualquer número de bits. Entretanto será sempre o complemento de 2 do número 1 com a quantidade de bits em que será representado. Resposta:

*`http://www.inf.ufsc.br/~dovicchi --- dovicchi@inf.ufsc.br`

c.1. Com 2 bits:

$$\begin{array}{rcr} 1_{10} & \rightarrow & \texttt{01} \\ \text{complemento} & \rightarrow & \texttt{10} \\ \text{compl. de 2 (soma 1)} & \rightarrow & \underline{\texttt{+1}} \\ -1_{10} & \rightarrow & \texttt{11} \end{array}$$

c.2. Com 4 bits:

$$\begin{array}{rcr} 1_{10} & \rightarrow & \texttt{0001} \\ \text{complemento} & \rightarrow & \texttt{1110} \\ \text{compl. de 2 (soma 1)} & \rightarrow & \underline{\texttt{+1}} \\ -1_{10} & \rightarrow & \texttt{1111} \end{array}$$

c.3. Com 8 bits:

$$\begin{array}{rcr} 1_{10} & \rightarrow & \texttt{00000001} \\ \text{complemento} & \rightarrow & \texttt{11111110} \\ \text{compl. de 2 (soma 1)} & \rightarrow & \underline{\texttt{+1}} \\ -1_{10} & \rightarrow & \texttt{11111111} \end{array}$$

d. +516

O número 516 sinalizado precisa de 11 bits para ser representado, pois com 10 bits podemos representar de -512 a 511 (1024 valores ou 2^{10} valores). Resposta: `01000000100`

e. -647

O mesmo para o -647, ou seja, tomamos o 647 com 11 bits e tiramos o complemento de 2. Resposta:

$$\begin{array}{rcr} 647_{10} & \rightarrow & \texttt{01010000111} \\ \text{complemento} & \rightarrow & \texttt{10101111000} \\ \text{compl. de 2 (soma 1)} & \rightarrow & \underline{\texttt{+1}} \\ -647_{10} & \rightarrow & \texttt{10101111001} \end{array}$$

2. Converta os números nas bases abaixo para números binários:

a. 45_8

Resposta: `100101`

Obs.: Quando a base a ser convertida para a base 2 for potência de 2, cada dígito pode ser representado em grupos desta potência. Assim, como $8 = 2^3$, cada dígito pode ser representado por binários com 3 bits: $4_{10} = 100_2$ e $5_{10} = 101_2$. Na base hexadecimal ($16 = 2^4$) cada dígito pode ser convertido para binários com 4 bits.

b. 36_5

Resposta: O número 36 na base 5 não existe, pois qualquer número na base 5 tem apenas os dígitos {`0,1,2,3,4`}.

c. 3211_4

Resposta: `11100101` (`11 10 01 01`)

d. $3BFA_{16}$

Resposta: `0011101111111010` (`0011 1011 1111 1010`)

e. $B21F_{16}$

Resposta: `1011001000011111`

3. Efetue as operações dos seguinte números binários puros:

a. $1011010111_2 + 1011010111_2$

Resposta: `10110101110`

b. $1000_2 + 1111_2$

Resposta: `10111`

c. $11011001_2 + 00100110_2$

Resposta: `11111111`

d. $1110111_2 + 1111_2 + 1001_2$

Resposta: `10001111`

e. $1111_2 + 111_2 + 11_2 + 1_2$

Resposta: `11010`

4. Efetue as operações na base 2 com os seguintes números com sinal:

a. $01100011 + 00001000$

Resposta: `01101011`

b. $01100011 + 01100100$

Resposta: `011000111`

Obs.: A soma destes dois números positivos resulta em um número positivo que precisa de um bit a mais para ser representado.

c. $10010001 + 11000101$

A soma de dois números negativos pode ser feita diretamente, pois se tirarmos o complemento de 2 de ambos, somá-los e tirarmos novamente o complemento de 2, temos o mesmo resultado. Entretanto, isto pode levar a erros quando for necessário usar mais bits para representar o resultado. A forma correta é tomar o complemento de 2 de cada operando, somá-los e tirar o complemento de 2 do resultado. Resposta: `101010110`

Obs.: Lembre-se que $-x + (-y) = -(x + y)$

d. $01001100 - 10010101$

Neste caso temos um número positivo menos um número negativo. Assim, $x - (-y) = x + y$. Então basta tirar o complemento de dois do número negativo para achar o seu valor positivo e somá-lo ao número positivo. Resposta: `010110111`

e. $11001110 - 10001000$

Neste caso temos $-x - (-y) = -x + y$. Como $|y| > |x|$, temos que encontrar o complemento de 2 de y e somar com x (já em complemento, o que significa fazer $y - x$). Resposta: `01000110`

5. Multiplique os números binários considerando o bit de sinal.

a. 01001000 × 00001011

```
     01001000
×    00001011
     --------
     01001000
    01001000
   00000000
  01001000
  -----------
  01100011000
```

b. 11010000 × 10001001

Trata-se da multiplicação de 2 números negativos. Então deve-se tirar o complemento de 2 de ambos e efetuar a operação normalmente (lembrando sempre de verificar a quantidade de bits necessária para representar o resultado). O resultado é positivo. Resposta: `01011001010000`

c. 11001100 × 00000011

Neste caso a multiplicação é entre um número negativo e um positivo. Temos, então, que tirar o complemento de 2 do número negativo, efetuar a multiplicação entre dois positivos e tornar a tirar o complemento de 2 do resultado, pois este é negativo. Resposta: `101100100`

d. 10000100 × 00001000

Resposta: `10000100000`

e. 01000000 × 00000100

Resposta: `0100000000`