UNIVERSIDADE ESTADUAL DO OESTE DO PARANÁ - UNIOESTE

CENTRO DE ENGENHARIAS E CIÊNCIAS EXATAS - CECE

CIÊNCIA DA COMPUTAÇÃO

DISCIPLINA: SISTEMAS DIGITAIS

DOCENTE: JORGE HABIB E ANTONIO HACHISUCA

ISABELA PIMENTEL LOEBEL

LISTA DE EXERCÍCIOS 1 -

SISTEMAS DE NUMERAÇÃO E CÓDIGOS

FOZ DO IGUAÇU,

2020.

# **Parte I - Livro Ronald J. Tocci**

2 Converta os seguintes números binários em decimais.

(b) 1001 0101

R: 149

(e)\* 1111 1111

R: 255

(f) 0110 1111

R: 111

(g)\* 11 1101 0111

R: 983

2.2 Converta os seguintes valores decimais em binários.

(b) 13 /2

1 6 /2

0 3 /2

↖ 1 1

R: 1101

(c)\* 189/2

1 94 /2

0 47 /2

1 23 /2

1 11 /2

1 5 /2

1 2 /2

↖ 0 1

R: 1011 1101

(d) 1000 /2

0 500 /2

0 250 /2

0 125 /2

1 62 /2

0 31 /2

1 15 /2

1 7 /2

1 3 /2

1 1

R: 11 1110 1000

(e)\* 77/2

1 38 /2

0 19 /2

1 9 /2

1 4 /2

0 2 /2

0 1

R: 100 1101

(f) 390 /2

0 195 /2

1 97 /2

1 48 /2

0 24 /2

0 12 /2

0 6 /2

0 3 /2

1 1

R: 1 1000 0110

(h) 2133 /2

1 1066 /2

0 533 /2

1 266 /2

0 133 /2

1 66 /2

0 33 /2

1 16 /2

0 8 /2

0 4 /2

0 2 /2

0 1

R: 1000 0101 0101

2.3 Qual é o maior valor decimal que pode ser representado por

(b) um número de 16 bits?

R: 1111 1111 1111 1111 = 65535

2.4 Converta cada número hexadecimal em seu equivalente decimal.

(a)\* 743

3\*160 = 3

4\*161 = 64

7\*162 = 1792

R: 1859

(b) 36

6\*160 = 6

3\*161 = 48

R: 54

(c)\* 37FD

D(13)\*160 = 13

F(15)\*161 = 240

7\*162 = 1792

3\*163 = 12288

R: 14333

(d) 2000

0\*160 = 0

0\*161 = 0

0\*162 = 0

2\*163 = 8192

R: 8192

(e)\* 165

5\*160 = 5

6\*161 = 96

1\*162 = 256

R: 357

(f) ABCD

D(13)\*160 = 13

C(12)\*161 = 192

B(11)\*162 = 2816

A(10)\*163 = 40960

R: 43981

2.5 Converta os números decimais em seu equivalente hexadecimal.

(c)\* 919/16

7 57 /16

9 3

R: 397

(d) 1024/16

0 64 /16

0 4

R: 400

(e)\* 771/16

3 48 /16

0 3

R: 303

(f) 2313 /16

9 144 /16

0 9

R: 909

2.9\* Quando um número decimal grande é convertido em binário, algumas vezes é mais fácil convertê-lo primeiro em hexadecimal e, então, em binário. Experimente esse procedimento para o número e compare-o com o procedimento usado no Problema 2.2(h).

*Passo 1: divisão por 16.*

2133 /16

5 133 /16

5 8

*Passo 2: conferir a tabela do livro.*

855

8 = 1000

5 = 0101

5 = 0101

R: 1000 0101 0101

2.10 Quantos dígitos hexadecimais são necessários para representar números decimais até 20.000? E até 40.000?

R: FFFF = 65535, sendo o maior número de 4 dígitos hexadecimais, podemos concluir que apenas 4 serão suficiente.

2.11 Converta os valores hexadecimais a seguir em decimais.

(b) 1A6

6\*160 = 6

A(10)\*161 = 160

1\*162 = 256

R: 422

(c)\* 37FD

D(13)\*160 = 13

F(15)\*161 = 240

7\*162 = 1792

3\*163 = 12288

R: 14333

(d) ABCD

D(13)\*160 = 13

C(12)\*161 = 192

B(11)\*162 = 2816

A(10)\*163 = 40960

R: 43981

(e)\* 000F

F(15)\*160 = 15

0\*161 = 0

0\*162 = 0

0\*163 = 0

R: 15

2.12 Converta os valores decimais a seguir em hexadecimais.

(d) 24 /16

8 1

R: 18

(e)\* 7245 /16

13 452 /16

4 28 /16

12 1

R: 1C4D

(f) 498 /16

2 31 /16

15 1

R: 1F2

(g)\* 25.619 /16

3 1601 /16

1 100 /16

4 6

R: 6413

2.13 Escreva o dígito hexa equivalente para os seguintes números binários de 4 bits na ordem em que foram escritos, sem fazer cálculos por escrito nem com a calculadora.

(a) 1001

R: 9

(b) 1101

R: 13

(c) 1000

R: 8

(d) 0000

R: 0

(e) 1111

R: F

2.14 Escreva o número binário de 4 bits para o equivalente dígito hexa, sem fazer cálculos por escrito nem com a calculadora.

(f) 3

R: 0011

(g) C

R: 1100

(h) B

R: 1011

(i) 9

R: 1001

(j) A

R: 1010

(k) 2

R: 0010

(l) F

R: 1111

2.15\* Qual é o maior valor que pode ser representado por três dígitos hexa?

R: FFF = 4095

2.17\* Relacione os números hexa, em sequência, de 280 a 2A0.

R: 281, 282, 283, 284, 285, 286, 287, 288, 289, 28A, 28B, 28C, 28D, 28E, 28F, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 29A, 29B, 29C, 29D, 29E, 29F, 2A0.

2.18 Quantos dígitos hexadecimais são necessários para representar os números decimais até 1 milhão? E até 4 milhões?

R: FFFFF = 1048575, 5 dígitos para representar 1 milhão, FFFFFF = 16777215, 6 dígitos para representar 4 milhões.

2.19 Codifique os números decimais a seguir em BCD.

(b) 962

9 = 1001

6 = 0110

2 = 0010

R: 1001 0110 0010

(c)\* 187

1 = 0001

8 = 1000

7 = 0111

R: 0001 1000 0111

(d) 6.727

6 = 0110

7 = 0111

2 = 0010

7 = 0111

R: 0110 0111 0010 0111

(e)\* 13

1 = 0001

3 = 0011

R: 0001 0011

(f) 529

5 = 0101

2 = 0010

9 = 1001

R:0101 0010 1001

(g)\* 89.627

8 = 1000

9 = 1001

6 = 0110

2 = 0010

7 = 0111

R: 1000 1001 0110 0010 0111

2.20 Quantos bits são necessários para representar os números decimais na faixa de 0 a 999 usando (a) o código binário puro? (b) E o código BCD?

R: Para binário puro são necessários 10 bits e para código BCD são necessários 12 bits.

2.21 Os números a seguir estão em BCD. Converta-os em decimal.

(b) 0001 1000 0100

R: 184

(c)\* 0110 1001 0101

R: 695

(d) 0111 0111 0111 0101

R: 7775

(e)\* 0100 1001 0010

R: 492

(f) 0101 0101 0101

R: 555

2.22 (c) Qual é o maior valor decimal codificado em BCD que pode ser representado com 3 bytes?

R: 1001 1001 1001 1001 1001 1001 = 999999 no total

2.23 (a) Consulte a Tabela 2.4. Qual é o nibble mais significativo do código ASCII para a letra X?

R: X - decimal = 88 - binário = 1011000, sendo os mais significativos 0101

(b) Quantos nibbles podem ser armazenados em uma palavra de 16 bits?

R: 16/4 = 4 nibbles

(c) Quantos bytes são necessários para formar uma palavra de 24 bits?

R: 3 bytes apenas

2.26 Os bytes a seguir (mostrados em hexa) representam o nome de uma pessoa do modo como foi armazenado na memória de um computador. Cada byte é um código em ASCII com um bit (MSB) anexado. Determine o nome da pessoa.

(a)\* 42 45 4E 20 53 4D 49 54 48

42 = 0100 0010 = B

45 = 0100 0101 = E

4E = 0100 1110 = N

20 = 0010 0000 =

53 = 0101 0011 = S

4D = 0100 1101 = M

49 = 0100 1001 = I

54 = 0101 0100 = T

48 = 0100 1000 = H

(b) 4A 6F 65 20 47 72 65 65 6E

4A = J

6F = O

65 = E

20 =

47 = G

72 = R

65 = E

65 = E

6E = N

2.27 Converta os seguintes números decimais para o código BCD e, em seguida, anexe um bit de paridade ímpar.

(c)\* 8884

R: 1 1000 1000 1000 0100

(d) 275

R: 1 0010 0111 0101

(e)\* 165

R: 0 0001 0110 0101

(f) 9201

R: 1 1001 0010 0000 0001

2.30 Faça as conversões a seguir. Em algumas, você pode querer experimentar diversos métodos para ver qual é mais prático. Por exemplo, a conversão de binário em decimal pode ser feita diretamente ou pode-se fazer uma conversão de binário em hexadecimal e, em seguida, de hexadecimal em decimal.

d) (1110101000100111)2 = 59943 10

e) (2497)10 = 9C1 16

f) (511)10 = 0101 0001 0001 (BCD)

g) (235)16 = EB 10

2.31\* Represente o valor decimal 37 em cada uma das seguintes formas:

(a) binário puro

R: 0010 0101

(b) BCD

R: 0011 0111

(c) hexa

R: 25

(d) ASCII (isto é, considere cada dígito um caractere)

R: 33 37 em hexa, 51 55 em decimal

2.32\* Preencha os espaços em branco com a(s) palavra(s) correta(s).

(b) A conversão de decimal em binário requer divisões sucessivas por 2 .

(c) No código BCD, cada dígito é convertido no equivalente binário de 4 bits.

(d) O código grey altera apenas um bit quando passamos de uma representação, no código, para a seguinte.

(e) Um transmissor anexa um bit de paridade aos bits do código para permitir ao receptor detectar ruídos.

2.34 Aplique uma operação de decremento a cada número

binário.

(a)\* 1100

R: 1011

(b) 10 1000

R: 10 0111

2.35 Escreva os números resultantes quando cada um dos

seguintes números é incrementado.

b) (9999)16

(999A)16

c) (0FFF)16

(1000)16

d) (2000)16

(2000)16

e) (9FF)16

(A00)16

f) (100A)16

(100B)16

2.37 Os endereços das posições de memória de um microcomputador são números binários que identificam cada posição da memória em que um byte é armazenado. O número de bits que constitui um endereço depende da quantidade de posições de memória. Visto que o número de bits pode ser muito grande, o endereço é especificado em hexa em vez de binário.

a) Se um microcomputador tem 20 bits de endereço, quantas posições diferentes de memória ele possui?

posições.

b) Quantos dígitos hexa são necessários para representar um endereço de uma posição de memória?

= 5

São necessários apenas 5 dígitos hexa.

c) Qual é o endereço, em hexa, da 256a posição da memória? (Observação: o primeiro endereço é sempre zero.)

posição.

2.38 Em um CD de áudio, o sinal de tensão de áudio é amostrado cerca de 44.000 vezes por segundo, e o valor de cada amostra é gravado na superfície do CD como um número binário. Em outras palavras, cada número binário gravado representa um único ponto da forma de onda do sinal de áudio.

a) Se os números binários têm uma extensão de 6 bits, quantos valores diferentes de tensão podem ser representados por um único número binário? Repita o cálculo para 8 e 10 bits.

b) Se forem usados 10 bits, quantos bits serão gravados no CD em 1 segundo?

bits por segundo.

c) Se um CD tem capacidade de armazenar 5 bilhões de bits, quantos segundos de áudio podem ser gravados quando forem utilizados números de 10 bits?

= 11363.6363636 segundos.

# **Parte II - Livro Thomas L. Floyd**

1.

c) 34

2. The decimal equivalent of 1000 is

d) 8

3.The binary number 11011101 is equal to the decimal number

b) 221

4. The decimal number 21 is equivalent to the binary number

a) 10101

5. The decimal number 250 is equivalent to the binary number

a) 11111010

6. The sum of 1111 + 1111 in binary equals

c) 11110

7. The difference of 1000 - 100 equals

c) 110

8. The 1’s complement of 11110000 is

c) 00001111

9. The 2’s complement of 11001100 is

b) 00110100

10. The decimal number 122 is expressed in the 2’s complement form as

b) 11111010

11. The decimal number -34 is expressed in the 2’s complement form as

c) 11011110

12. A single-precision floating-point binary number has a total of

d) 32 bits

13. In the 2’s complement form, the binary number 10010011 is equal to the decimal number

d) -109

14. The binary number 101100111001010100001 can be written in octal as

b) 5471241

15. The binary number 10001101010001101111 can be written in hexadecimal as

c) 8D46F

16. The binary number for F7A9 is

a) 1111011110101001

17. The BCD number for decimal 473 is

c) 0100 0111 0011

18. Refer to Table 2–7. The command STOP in ASCII is

a) 1010011101010010011111010000

19. The code that has an even-parity error is

d) 1110111

15. Add the binary numbers

a) 10+ 10

100

b) 10+11

101

c) 100+111

1011

d) 111+101

1100

e) 1111+ 111

010110

f) 1111+1111

011110

16. Use direct subtraction on the following binary numbers:

a) 10 - 1

1

b) 100 - 11

1

c) 110 - 100

010

d) 1111 - 11

01100

e) 1101 - 101

1000

f) 110000 - 1111

010000

19. What are two ways of representing zero in 1’s complement form?

Neste complemento, há dois métodos de representação do zero:

→ Zero positivo: dado pelo número binário: 000000;

→ Zero negativo: dado pelo número binário 111111.

20. How is zero represented in 2’s complement form?

Pelo método convencional: 00000.

22. Determine the 2’s complement of each binary number using either method:

a) 11

10

b) 110

101

c) 1010

1001

d) 1001

1001

e) 101010

101001

f) 11001

11000

g) 11001100

11001011

h) 11000111

11000110

23. Express each decimal number in binary as an 8-bit sign-magnitude number:

a) 29

11101

b) -85

-1010101

25. Express each decimal number as an 8-bit number in the 2’s complement form:

a)12

0000000000001100

b) -68

1111111110111100

26. Determine the decimal value of each signed binary number in the sign-magnitude form:

a) 10011001

153

b) 01110100

116

c) 10111111

191

28. Determine the decimal value of each signed binary number in the 2’s complement form:

a) 10011001

-103

b) 01110100

116

c) 10111111

-65

29. Express each of the following sign-magnitude binary numbers in single-precision floating point format:

a) 0111110000101011

124

b) 100110000011000

-48

30. Determine the values of the following single-precision floating-point numbers:

a) 1 10000001 01001001110001000000000

-5.15258

b) 0 11001100 10000111110100100000000

2.31289888882e+23

31. Convert each pair of decimal numbers to binary and add using the 2’s complement form:

a) 33 and 15

100001 + 1111 = 110000

b) 56 and 227

111000 + 11100011 = 100011011

c) 246 and 25

11110110 + 11001 = 100001111

d) 2110 and 284

100000111110 + 100011100 =100101011010

32. Perform each addition in the 2’s complement form:

a) 00010110 + 00110011

1001001

b) 01110000 + 10101111

100011111

33. Perform each addition in the 2’s complement form:

a)10001100 + 00111001

11000101

b)11011001 + 111100111

1011000000

34. Perform each subtraction in the 2’s complement form:

a) 00110011 - 00010000

100011

b) 01100101 - 11101000

111101111101

47. Convert each of the following decimal numbers to 8421 BCD:

a) 10

0001 0000

b) 13

0001 0011

c) 18

0001 1000

d) 21

0010 0001

49. Convert the following decimal numbers to BCD:

a) 104

0001 0000 0100

b) 128

0001 0010 1000

c) 132

0001 0011 0010

d) 150

0001 0101 0000

50. Convert each of the BCD numbers to decimal:

a) 0001

1

b) 0110

3

c) 1001

9

d) 00011000

1 8

e) 00011001

1 9

51. Convert each of the BCD numbers to decimal:

a) 10000000

80

b) 001000110111

237

c) 001101000110

346

h) 0001011010000011

1683

i) 1001000000011000

9019

j) 0110011001100111

6667

52. Add the following BCD numbers:

a) 0010 + 0001

0011

f) 01100100 + 00110011

10010111

g)01000000 + 0100011

01100011

h) 1000101 + 00010011

10011000

54. Convert each pair of decimal numbers to BCD, and add as indicated:

a) 4 + 3

0100 + 0011 = 0111

b) 5 + 2

0101+ 0010 = 0111

g) 113 + 101

0001 0001 0011 + 0001 0000 0001 = 0010 0001 0011

h) 295 +157

0010 1001 0101 + 0001 0101 0111 = 0100 0101 0010

55. In a certain application a 4-bit binary sequence cycles from 1111 to 0000 periodically. There are four bit changes, and because of circuit delays, these changes may not occur at the same instant. For example, if the LSB changes first, the number will appear as 1110 during the transition from 1111 to 0000 and may be misinterpreted by the system. Illustrate how the Gray code avoids this problem.

O código Gray consegue evitar esse problema utilizando uma contagem diferente do binário puro, onde apenas um bit muda por a cada transição.

56. Convert each binary number to Gray code:

a) 11011

10110

b) 1001010

1101111

c) 1111011101110

1000110011001

57. Convert each Gray code to binary:

a) 1010

1100

b) 00010

00011

c) 11000010001

10000011110

60. Decode the following ASCII coded message:

1001000 1100101 1101100 1101100 1101111 0101110

100000 1001000 1101111 1110111 0100000 1100001

1110010 1100101 0100000 1111001 1101111 1110101 111111

Hello. Hoo arr yoo Del

62. Convert the following statement to ASCII:

30 INPUT A, B

0110011 0110000 0100000 0001111 0100000 1000001 0101100 0100000 1000010

63. Determine which of the following even parity codes are in error:

b) 011101010

64. Determine which of the following odd parity codes are in error:

a) 11110110

c) 01010101010101010

65. Attach the proper even parity bit to each of the following bytes of data:

a) 10100100

1 10100100

b) 00001001

0 00001001

c) 11111110

1 11111110