Checked by:
- Sandra Aw, Hang Tu
- Kaitelin Rivera, Dominic Ruiz

Name:Eli Griffiths
Name:Sebastian Cedano

Data Storage – Worksheet II

Two's complement conversions

1. Convert each of the following two's complement representations to its equivalent base ten form:
   a. 00011 => 3
   b. 01111 => 15
   c. 11100 => -4
   d. 11010 => -6
   e. 00000 => 0
   f. 10000 => -16
2. Convert each of the following base ten representations to its equivalent two's complement form using patterns of 8 bits:
   a. 6 => 00000110
   b. -6 => 11111010
   c. -17 => 11101111
   d. 13 => 00001100
   e. -1 => 11111111
   f. 0 => 00000000
3. Suppose the following bit patterns represent values stored in two's complement notation. Find the two's complement representation of the negative of each value:
   a. 00000001 => 11111111
   b. 01010101 => 10101011
   c. 11111100 => 00000100
   d. 11111110 => 10101011
   e. 00000000 => 00000000
   f. 01111111 => 10000001
4. Suppose a machine stores numbers in two's complement notation. What are the largest and smallest numbers that can be stored if the machine uses bit patterns of the following lengths?
   a. four => [-8, 7]
   b. six => [-32, 31]
   c. eight => [-128, 127]
5. In the following problems, each bit pattern represents a value stored in two's complement notation. Find the answer to each problem in two's complement notation by performing the addition process described in the powerpoint slides. Then check your work by translating the problem and your answer into base ten notation.
   a. 0101+0010 = 0111 = 7

   b. 0011+0001 = 0100 = 4
   c. 0101+1010 = 1111 = -1
   d. 1110+0011 = 0001 = 1
   e. 1010+1110 = 1000 = -8
6. Solve each of the following problems in two's complement notation, but this time watch for overflow and indicate which answers are incorrect because of this phenomenon.
   a. 0100+0011 = 0111 = 7
   b. 0101+0110 = OVERFLOW (11)
   c. 1010+1010 = 1000 = -8
   d. 1010+0111 = 0001 = 1
   e. 0111+0001 = OVERFLOW (8)
7. Translate each of the following problems from base ten notation into two's complement notation using bit patterns of length four, then convert each problem to an equivalent addition problem (as a machine might do), and perform the addition. Check your answers by converting them back to base ten notation.
   a. (-6)-(-1) = (-6) + 1 = 1010 + 0001 = 1011 = -5
   b. 3-2 = 3 + (-2) = 0011 + 1110 = 0001 = 1
   c. 4-6 = 4 + (-6) = 0100 + 1010 = 1110 = -2
   d. 2-(-4) = 2 + 4 = 0010 + 0100 = 0110 = 6
   e. 1-5 = 1 + (-5) = 0001 + 1011 = 1100 = -4
8. Can overflow ever occur when values are added in two's complement notation with one value positive and the other negative? Explain your answer.

   Since the addition of a negative and positive number results in a smaller number, there cannot be an overflow error as the result stays bounded between the two numbers.

Excess Notation conversions

9. Convert each of the following excess eight representations to its equivalent base ten form without referring to the table in the slides:
   a. 1110 => 6
   b. 0111 => -1
   c. 1000 => 0
   d. 0010 => -6
   e. 0000 => -8
   f. 1001 => 1
10. Convert each of the following base ten representations to its equivalent excess eight form without referring to the table in the slides:
   a. 5 => 1101
   b. -5 => 0011

c. 3 => 1011
d. 0 => 1000
e. 7 => 1111
f. -8 => 0000

11. Can the value 9 be represented in excess eight notation? What about representing 6 in excess four notation? Explain your answer. No, there are not enough bits to represent 9 in excess 8 notation. There aren't enough bits for 6 in excess 4 either.

Floating-Point Notation Conversions

12. Decode the following bit patterns using the floating-point format discussed in the slides:
    a. 01001010 => 5/8
    b. 01101101 => 3 1/4
    c. 00111001 => 9/32
    d. 11011100 => -1 1/2
    e. 10101011 => -11/64
13. Encode the following values into the floating-point format discussed in the slides. Indicate the occurrence of truncation errors.
    a. $2\frac{3}{4}$ => 01101011
    b. 5 ¼ => 01111010 lost one bit in the fourths place
    c. $\frac{3}{4}$ => 01001100
    d. $-3\frac{1}{2}$ => 11101110
    e. $-4\frac{3}{8}$ => 11111000 lost two bits in the fourths and eighths place