# Exercise 1 Problems

Do NOT use a calculator on this assignment. You will not be allowed to use one on the test or quizzes.

1. Convert the following binary numbers to decimal.
   1. 100112
   2. 011012
   3. 100011102
   4. 110111012
   5. 1101011112
   6. 1010101012
   7. 100112c
   8. 011112c
   9. 101011012c
   10. 011010112c
   11. 1000111112c
   12. 0110110112c
2. Convert the following decimal numbers to unsigned 5-bit binary, 8-bit binary, and 9-bit binary or state that a conversion is not possible (i.e. the decimal number cannot be represented with the given number of bits). Represent your answers in binary then convert the binary representations to hexadecimal.
   1. 910
   2. 1910
   3. 4510
   4. 6710
   5. 8810
   6. -9010
   7. 12410
   8. 20110
   9. 30010
   10. 40510
3. Convert the following decimal numbers to signed 5-bit binary, 8-bit binary, and 9-bit binary or state that a conversion is not possible (i.e. the decimal number cannot be represented with the given number of bits). Represent your answers in binary then convert the binary representations to hexadecimal.
   1. -1210
   2. -910
   3. 1110
   4. -5010
   5. -9910
   6. 7810
   7. 13010
   8. 10310
   9. -23310
   10. -33310
4. Convert the following decimal numbers directly to hexadecimal.
   1. 2710
   2. 4510
   3. 5810
   4. 8310
   5. 12010
   6. 18010
   7. 21010
   8. 23510
   9. 24010
   10. 25510
5. For each of the systems listed, convert the hexadecimal value directly to decimal and directly to binary.
   1. 5-bit unsigned binary
      1. 0A16
      2. 1F16
      3. 0E16
      4. 1716
   2. 8-bit unsigned binary
      1. ED16
      2. 8F16
      3. B416
      4. 6A16
   3. 9-bit unsigned binary
      1. 0A716
      2. 07916
      3. 10116
      4. 15516
   4. 5-bit signed binary
      1. 0A16
      2. 1A16
      3. 1F16
      4. 0816
   5. 8-bit signed binary
      1. AE16
      2. 7F16
      3. 5D16
      4. CC16
   6. 9-bit signed binary
      1. 0AE16
      2. 1AE16
      3. 08A16
      4. 1FF16
6. Negate the following binary numbers.
   1. 000012C
   2. 100112C
   3. 011012C
   4. 001101012C
   5. 010010002C
   6. 000000002C
   7. 1001010002C
   8. 0111101112C
   9. 100000002C
   10. 011110102
7. Extend the following numbers to 16 bits. Then, convert both the value listed and the 16-bit answer to hexadecimal.
   1. 010012
   2. 101112
   3. 101112c
   4. 001101012
   5. 001101012c
   6. 101100112
   7. 1000100112c
   8. 0110111002c
8. Truncate each value listed to 9, 8, and 5 bits or state that the truncation is not valid.
   1. 0000 0000 0001 10112
   2. 1111 1111 1110 01112c
   3. 1111 1111 1011 10002c
   4. 0000 0000 1010 00112c
   5. 1111 1100 0011 11112c
9. For each problem, generate the result using the correct number of bits and state whether or not overflow occurs.
   1. 001102 + 101112
   2. 001102 - 101002
   3. 110002c + 111112c
   4. 001112c - 101012c
   5. 110012c + 110102c
   6. 011102c - 101112c
   7. 011010112 + 100110112
   8. 010000002 + 011100002
   9. 110011002c + 110011112c
   10. 011101012c + 101111112c
   11. 011011012 - 011011112
   12. 100111102 - 011000002
   13. 100111102c - 011000002c
   14. 101101012c - 110010112c
   15. 010100002c + 011100002c
   16. 010100002c - 011100002c
   17. 110001102 + 11102
   18. 010111012 - 01012
   19. 011100002c + 11112c
   20. 111101102c - 01112c