* IEEE (Institute for Electrical and Electronic Engineers) Standard 754 1985
  + Storing a floating point number on the computer using a common standard. Devote 64 bits of storage for a floating-point number with
    - 1 bit for the sign
    - 2 bits for the exponent
    - 52 bits for the mantissa
* Ten’s compliment
  + Use vertical number scale
* Representing negative values
  + Two’s complement
    - If positive, left most bit is 0; if negative left most is 1
  + Addition and subtraction are the same
* Number overflow
  + If each value is stored using eight bits, adding 127 to 3 overflows
    - Problems occur when mapping an infinite world onto a finite machine
* Example
  + 16-bit bank goes up to 37,768 bits
* Real numbers
  + A number with a whole part and a fractional part
  + 20.25 to (B2) => 10100.01  
    0.25 \* 2 = 0.50  
    0.50 \* 2 = 1.00
  + 2.625 to (B2) => 10.101  
    0.625 x 2 = 1.25  
    0.25 x 2 = 0.50  
    0.50 x 2 = 1.00
  + 0.40625 => 0.01101  
    0.40625 x2 = 0.8125  
    0.8125 x 2 = 1.625  
    0.625 x 2 = 1.25  
    0.25 x 2 = 0.5  
    0.5 x 2 = 1
  + Decimal point is actually the **radix point**. Positions to the right of the radix point in binary are
    - 2-1 (one half)
    - 2-2 (one quarter)
    - 2-3 (one eighth)
  + A real value in base 10 can be defined by the following formula  
    sign \* mantissa \* 10 exp
* Scientific notation
  + A form of floating point representation in which the decmail point is kept ot the right of the leftmost digit
  + 12001.32708 = 1.200132708E+4
  + 123.332 = 1.23332 E+2
  + 0.0034 = 3.4E-3
* Character set
  + A list of characters and the codes used to represent each one
  + Computer manufacturers agreed to standardize
  + Sets
    - ASCII (main) = American Standards Code for Information Interchange
      * Originally allowed for 128 unique characters
      * Was extended so that all 8-bits could be used
      * Now max 256 characters
    - ANSI (main) = American National Standards Institute
    - ECPSID?
    - Unicode
  + Example
    - Joe College  
      74,111,101,32,67,111,108,108,101,103,101 (ASCII Code)  
      01001010, 101101111, 01100101, 00100000 (Binary)
  + Extended ASCII is not enough for international use
  + Unicode mapping uses 16 bits per character
    - Is a superset of ASCII
    - The first 256 characters respond to the ASCII set
* Text compression
  + Assigning 16 bits to each character in a document uses too much file space
  + Techniques
    - Keyword encoding
    - Run-length encoding
    - Huffman encoding
* Keyword encoding
  + Replace frequently used words with a single character
  + Not standardized
  + Other notes/rules
    - Encoding characters cannot be a part of the original text.
    - Upper case/lower case issue – it is case sensitive
    - No gain in encoding words such as "a" or "I"