Data Types and Information Representation

9-12-2003

Opening Discussion

- What did we talk about last class?
- Do you have any questions about the assignment? Are there any significant problems that people are running into? Don't spend too long banging your head against a wall. Ask questions and feel free to "walk away" for a while.
- Who can describe the decimal number system that we use?

Computer Information 0/1

- As you know, computers store everything as a series of bits where each bit can be on or off, we represent this as 1 or 0.
- Terms
  - Byte - 8 bits
  - Word, HalfWord, and Dword - Varies by machine, but today a "standard word" would be considered to have 32 bits in it.
  - Nibble - Half a byte. This is never used now.

Binary Numbers

- You are used to the decimal number system where we have 10 digits, 0-9, and each digit going left represents groupings 10 times larger that the previous digit.
  - 1, 10, 100, 1000, …
- In binary the same is true, but the base is 2 so we have only two digits and it position goes up by a factor of two.
  - 1, 2, 4, 8, 16, 32, 64, …

Conversion to and from Binary

- To convert to binary you begin by finding the largest power of two that is smaller than the number. Put a one in the digit for that power, subtract it from the number and repeat.
  - 29=16+13=16+8+5=16+8+4+1=11101
- To get back to decimal just add up the values of the powers of 2 where the bit is one.

Hexadecimal Numbers

- Also common when working with computers are hex numbers. These us a base 16 number system which has 0-9 and A-F. The same general rules apply.
- Because 16 is a power of 2 we can easily convert between the two by grouping bits into groups of 4. 0000=0, 0001=1, 0010=2, … 1110=E, 1111=F.
- Each hex digit is a nibble and two make a byte.

Octal Numbers

- You will also occasionally see numbers in base-8 as well. Again we have a power of 2, but now the bits are grouped by 3.

Binary Addition

- Adding in binary is pretty easy, even easier than in decimal.
- You can write the numbers one above the other and perform "long addition" with carrying. If both have a zero the result is zero. If one has a one, the result is one. If both have a one, the result is zero and you carry one to the next digit.
- Let's look at small examples.

Negative Values

- How do we represent a negative value when we have only 0/1? Your book mentions a sign bit, that isn't really how it is done with integers for many reasons.
- Instead, we use what is called 2s compliment numbers. The idea is that a number plus its negation should always equal zero.
- Let's explore this idea on a 4 bit number.

Minute Essay

- Convert 187 to binary then write it as a hex number and an octal number. Remember the shortcuts for the last two.
- Assignment #1 is due by the end of today. I have received submissions from many of you. I'm guessing that they don't take all that long once you start them.
- Quiz #1 is Monday.