COS 301 Fall 2012

Homework #4 Due Thursday Nov 8

For the questions involving internal numeric representations, you may be able to find some answers by inspecting variables in a debugger. Others may require some web or textbook research.

1. (½ point each) Show internal representations in hexadecimal for the integer value 771 as
   a) Unsigned binary 32-bits
   b) a 32-bit IEEE float
   c) a 64-bit IEEE float
   d) binary coded decimal (unpacked)
   e) binary coded decimal (packed)

2. (½ point) Show the number 53 in 11-bit binary using a bias-1023 or excess 1023 representation (this is how exponents in doubles are stored)

3. (½ point each) Show the number -151 in hexadecimal, using 16-bit integers, in the following systems:
   a) Two's complement (the most commonly used representation)
   b) Signed-magnitude

4. (1 point) Consider the following C declarations:

```
unsigned char k = 2 ;    /* internally 0x02 */
char j = -2;             /* internally 0xFE */
```

What is the result of evaluating

```
char mystery = k < j ? 1 : 0;
```

and why?

For problems 5 and 6 you can make the simplifying assumption that integers are 4 bytes.

5.(4 points) Consider a three dimensional array declared with

```
int x3[5][5][5]
```

in C and

```
INTEGER x3(5,5,5)
```

in Fortran.

Assuming contiguous allocation of a block of memory, write an access function (in pseudocode or real code) that will yield the address of the element at index i,j,k for each language (**`x3[i][j][k] = t`** or **`x3(i,j,k) = t`**). Be sure to consider storage order and lower bounds for each language.

6. (1 point) Consider the above C array implemented using pointers. A 2 dimensional array would consist of an vector of pointers to vectors (one dimensional arrays) and a three dimensional array is a vector of pointers to the vector of pointers to vectors. Assuming that **`&x3`** yields the base address of x3, write a single expression using the dereferencing operator * that yields **`&x3[i][j][k]`** (the address of the element at i,j,k).