LAB 2: Variable Representations

Objectives:

a) Understand how integers, floating point numbers and character data types are stored in the computer.
b) Introduce some syntax of a low level language.

1) Background: In this lab you will investigate the different ways that integers, floating point numbers and character variables are stored and processed in the computer.

2) Exercises:
   a) Go to the Frances-A site and type the following code into Frances-A.

```
int main(){
        int   n=5;
        char c='5';
        float y=5;
}
```

      i) Write the assembly code that stores 5 into n.
      ii) Write the assembly code that stores '5' into c.
      iii) Write the assembly code that stores 5 into y.
         (1) Why do you think the float is differently than the int and char?
         (2) This is stored in the IEEE 754 standard. Convert it from this representation to its decimal value. Show all work.
   b) Now step through the program.
      i) How many bytes does the integer occupy?
         (1) What is the address of and value stored in the memory location.
      ii) How many bytes does the char occupy?
         (1) What is the address of and value stored in the memory location.
         (2) What was stored in this address prior to execution of the instruction storing the char?
      iii) How many bytes does the float occupy?
         (1) What is the address of and value stored in the memory location.
   c) What are the *mov*, *movb*, and *movl* instructions? Why are they different?
   d) What is the purpose of the *sub* and *add* instructions that occur before and after these *mov* instructions?
   e) Now replace the line *char c;* with the line *char c[5] = {'1', '2', '3', '4', '5'};*
      i) Step through the program.
      ii) How is this array stored? Write the addresses and value in the memory locations prior to moving any data and then after each *mov* associated with the array.