Tutorial 8

Question 1: (10)

a) When a variable of type pointer to a pointer to a pointer to 8 bits of signed data is defined, how much space is memory is allocated for the variable? Explain why. (2)

b) Could this be different for other architectures? Explain. (2)

c) Explain what the reference and dereference operators do, and which data types they are defined on. (4)

d) If we call a function with the name of a variable as an argument, like:
`myFunction(myVariable);`
what is actually passed into the function? (1)

e) If we have a variable whose value we want to be modified by a function, what should be passed into the function as an argument? (1)

Question 2: (6)

a) Assuming element 0 of an array of 32 bit numbers is located at address 0x2000 0124, what is the effective address of element 42? (2)

b) If we allocate an array of 123 elements, what is the largest index which we should access to ensure that we only access elements in the array? (1)

c) Assume we have an array type with the name `foo`. Apparently we can ascertain the number of elements allocated for the array with `sizeof(foo)/sizeof(foo[0])`. Explain why this works. (2)

d) Assume we have an array type with name `foo`. Write an expression which will give us the effective address of element 42 of the array. (1)

Question 3: (6)

a) For our LCD screen, we attached a few components to the E line. Why do we need these and what do they do? (3)

b) Two initialisation commands are sent to the LCD. What do exactly each of these commands do? (3)