Course week #2

This week is a continuation of learning how to develop C-programs.

Goals for this week:

- You should be able to use:
- General C-structures
- Function with parameters
- Arrays
- Pointers

Course week #3

The overall aim of this week is to introduce you to low level programming which includes reading I/O ports and manage bits and bytes. To give you an opportunity to practice this without a lot of system related details to learn we will use a IDE for a Motorola MC12 CPU based system. The IDE includes a very easy to use simulator that allow us to test simple C-programs interacting with different I/O devices. The IDE is called XCC 12 and you can download and install it on your own computer as free-ware.

You find the installation program (only for Windows) in the Documents archive. In the same catalogue you will find a short Quick guide for start using the IDE.

Goals for the week:

You should know how to write C-programs for

- read and write to memory mapped I/O ports
- manage individual bits.

To be read
Intro to XCC12 develop environment
Introduction to I/O principles , Wikipedia
Introduction to interrupt principles , Wikipedia

Additional about exercises:
The exercises for course week #3 focus on managing of bits and low level C-programming.
For some of them you need to use the XCC12 IDE for a MC12S CPU based system.
As a windows user you can install the IDE XCC12 and use it for solving the exercises needed the environment.
If you not have access to a windows computer you can find the program at all Windows computers in the house Jupiter floor 3 or 0

Course week #4

This week you will continue your learning of more general C-programming.

Goals for the week:

You should know how to write C-programs using

- Structured Data Types
- Enums
- Dynamic memory
- Lists
- Files

To be read

It is recomended to study the lecture presentations and the distributed program files for lecture #4_1 in the Directory.

Read also: http://en.wikipedia.org/wiki/Memory_leak

About memory leak (simple ex of a memory leak). Read also very short the information for the tool valgrind (a Linux based tool) that you find a link to in the text of memory leaks.