



An Introduction to HPC and Scientific Computing

Wes Armour

Oxford e-Research Centre,
Department of Engineering Science

Oxford e-Research Centre

www.oerc.ox.ac.uk

Aims and learning outcomes

The aims of this CWM are to introduce you to scientific computing and High Performance computing (HPC).

It's more important that you pick up the basics of computing and programming during the week, because these are the building blocks for everything else.

This CWM isn't designed to turn you into a world class HPC programmer, that takes years.

This CWM is designed to give you the skills to continue to learn in this area and for you to have the ability to write your own computer codes and tackle basic problems.

Assessment for this course will focus on the final two practical sessions in the latter half of the week. The aim of the assessment is for you to demonstrate that you've picked up the basics from this course.

The assessment will be light because I'm keen for you to focus on the content rather than worrying about the assessment.

In all I hope you will find this a fun and interesting weeklong introduction to HPC and Scientific Computing!

The use of LLMs (ChatGPT, Copilot,...)

LLMs can be incredibly useful coding tools.

However,... if you don't understand the code they produce, then you have no hope of debugging it or knowing if it will produce the correct output.

Moreover, if you can't understand the code, you have no way of knowing whether you are about to execute code that poses a security threat to you, your machine or the infrastructure you are using.

So, let's focus on learning how to code first.

(Note: we can tell if an LLM has been used to write code, we will deduct marks if you use an LLM)

Locations and Timetable

Locations

Lectures – these will be in **LR6** and slides available through git / canvas.

Practical sessions will in **Linux Lab B** (thom 6th floor) and available through git (more to come on that).

Timetable – Approximately....

10:00 - 11:00 Morning lecture
11:00 - 11:30 break
11:30 - 13:00 Morning practical

13:00 - 14:00 lunch

14:00 - 15:00 Afternoon lecture
15:00 - 15:30 break
15:30 - 17:00 Afternoon practical

Lectures, demo's and practical sessions will be delivered by:

Wes Armour, Yishun Lu, Ollie Ogden, Radostin Stoyanov and Jack White.

Please, please, please do leave feedback for us (good or bad) <https://bit.ly/OXUNICWM> 😊

Lectures

Monday - Here we have three lectures to begin with and finish with a practical session, this is because we'll need to introduce you to several different topics before you can complete a meaningful practical.

Morning lecture:	Introduction to computer architectures.
Morning lecture:	Introduction to the C programming language.
Afternoon demo:	Introduction to Linux, compilers and build systems.

Tuesday

Morning demo / lecture:	Using repositories and good coding practices.
Afternoon lecture:	A deeper dive into C programming.

Wednesday Morning

Morning lecture:	How to multi-task on CPUs using OpenMP.
------------------	---

Thursday

Morning lecture:	An introduction to GPUs and how to use them.
Afternoon lecture:	Guest Lectures: Provided by NVIDIA
Second afternoon lecture:	An introduction to the CUDA programming language.

Friday

Morning lecture:	Scientific Computing using the CUDA programming language.
------------------	---

Practical Sessions

Monday - Here we have one practical in the afternoon.

Afternoon Practical: Linux, compiling C code and using Make.

Tuesday

Morning Practical: Practical examples of using repositories for your projects.

Afternoon Practical: Practical examples using the C programming language.

Wednesday Morning

Morning Practical: Practical examples of using OpenMP on CPUs.

Thursday

Morning Practical: Practical examples of using GPUs for science and engineering.

Friday

Morning Practical: Examples of CUDA programming.

Afternoon Practical: Assignment (email assignment to wes.armour@eng.ox.ac.uk **AT 17:00!**).

Assignment

**The assignment is a fun thing, but you do get some marks for it...
So please do complete the assignment!**

The marking scheme is as follows:

A total of 9 marks.

4 marks will be given for attendance, 5 marks for assignment work.

Assignment marks will be given for:

Good coding practices	- 2 marks.
Using a build system	- 1 mark.
Correct use of C/CUDA	- 1 mark.
Working code	- 1 mark.

And remember – please do send feedback!

<https://bit.ly/OXUNICWM>