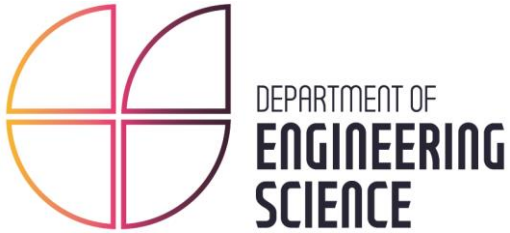

An Introduction to HPC and Scientific Computing

Wes Armour

Oxford e-Research Centre,
Department of Engineering Science

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 that's 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 week long introduction to HPC and Scientific Computing!

Locations and Timetable

Locations

Lectures will be in LR6
Practical sessions will be in the Linux Lab

Timetable

09:30 - 10:30 Morning lecture
10:30 - 11:00 break
11:00 - 12:30 Morning practical

12:30 - 13:30 lunch

13:30 - 14:30 Afternoon lecture
14:30 - 15:00 break
15:00 - 16:30 Afternoon practical

Lectures will be delivered by Wes Armour, Ian Bush, Karel Adamek.

Practical's supervised by Wes Armour, Ian Bush, Karel Adamek, Ania Brown and Jan Novotny.

On-line feedback form: http://bit.ly/OXUNICWM please, please, please do complete ☺

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 lecture:	Introduction to Linux, compilers and build systems.

Tuesday

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

Wednesday afternoon

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

Thursday

Morning lecture:	An introduction to GPUs and how to use them.
Afternoon lecture:	An introduction to the CUDA programming language.

Friday

Morning lecture:	Scientific Computing using the CUDA programming language.
Afternoon lecture:	***Guest Lecture: Deep Learning and challenges of scale - Adam Grzywaczewski (NVIDIA).***

UNIVERSITY OF OXFORD

Oxford e-Research Centre

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 Afternoon

Afternoon Practical:	Practical examples of using OpenMP on CPUs.

Thursday

Morning Practical:	Practical examples of using GPUs for science and engineering.
Afternoon Practical:	Examples of CUDA programming.

Friday

Morning Practical:	Assignment.
Afternoon Practical:	Finishing up (email assignment to wes.armour@eng.ox.ac.uk).

NVIDIA Guest lecture

Abstract

Majority of interesting problems tackled by industry are fairly complex. Where it is relatively easy to build an early POC of a system it takes a huge amount of effort to build a solution meeting all of your functional as well as non-functional requirements. For example its fairly straightforward to build a POC Self Driving Vehicle that will drive across a small number of streets with human supervision. On the other hand building a Self-Driving Car which a robust and safe is an engineering feet requiring petabytes of data for training and validation. In this talk we will tackle some of the key challenges of building complex Deep Learning based systems with a primary focus on scalability of the training process.

Bio

Dr Adam Grzywaczewski is a deep learning solution architect at NVIDIA, where his primary responsibility is to support a wide range of customers in delivery of their deep learning solutions. Adam is an applied research scientist specialising in machine learning with a background in deep learning and system architecture. Previously, he was responsible for building up the UK government's machine-learning capabilities while at Capgemini and worked in the Jaguar Land Rover Research Centre, where he was responsible for a variety of internal and external projects and contributed to the self-learning car portfolio.

Assignment

Course marks

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.