PHYS703

Computational Science

S1 Day 2015

Dept of Physics and Astronomy

Contents

General Information

Unit convenor and teaching staff
Unit Convenor
Jason Twamley
jason.twamley@mq.edu.au
Contact via jason.twamley@mq.edu.au
E6B 2.612
Thursdays 10--11am; other times by appointment.

Lectuer
David Spence
david.spence@mq.edu.au
Contact via 9850-8973
E6B 2.709

David Spence
david.spence@mq.edu.au

Credit points
4

Prerequisites
Admission to MRes

Corequisites

Co-badged status

Unit description
Computational techniques are a critical aspect of modern physics, science and engineering. They sit apart from theoretical and experimental physics but borrow characteristics from both. The aim is to turn a computer into a virtual laboratory for research, that allows breakthroughs and insights from what would otherwise be intractable problems by analytical methods. This unit focuses specifically on the computational techniques for solving problems in physics, engineering and science in general. It is not a course in programming though a low level of programming ability will be required to practice the techniques. Topics to be covered: o Introduction to Python and the Python scientific environment. o ODEs: Euler, Runge-Kutta and adaptive techniques, examining accuracy and stability with examples drawn from planetary science and chaotic systems. o Spectral methods: systems of linear equations, spectral analysis and analysis of normal modes. o PDEs: Initial and boundary conditions, discretisation. Relaxation and implicit schemes. Examples of Poisson, diffusion and wave equations. o Monte Carlo methods: random numbers, Monte Carlo integration, random walks, Metropolis algorithm. Examples of Ising model and phase transitions. o Convex optimisation: convex sets and functions, optimisation problems, linear and quadratic programming, duality.

Important Academic Dates

Information about important academic dates including deadlines for withdrawing from units are available at https://www.mq.edu.au/study/calendar-of-dates

Learning Outcomes

On successful completion of this unit, you will be able to:

This unit is designed to give the students a tool set of computational techniques that will be useful in their research and future careers. Students will be able to:
understand the framework of techniques and which techniques would be most applicable to a given problem.
demonstrate an understanding of the methods covered.
apply their knowledge to solve computational problems, creating and testing the necessary routines.
understand how to analyse and evaluate the accuracy of a computational routine.
develop skills in basic programming, code organisation
understand at an introductory level key concepts in modern scientific programming - classes, parallel techniques

Assessment Tasks

Name	Weighting	Due
Assignments	25%	TBA
In-Lab Assessment	25%	Weekly
Project	25%	Week 12
Final Examination	25%	Exam Weeks

Assignments

Due: **TBA**
Weighting: **25%**

The assignments will comprise of 3-4 questions designed to engage the students with

the material as it's covered. The difficulty of the questions will be set so that the assignment would take on

average around 7 hours to complete ; mix of theory and programming work

On successful completion you will be able to:

- This unit is designed to give the students a tool set of computational techniques that will be useful in their research and future careers. Students will be able to:
- understand the framework of techniques and which techniques would be most applicable to a given problem.
- demonstrate an understanding of the methods covered.
- apply their knowledge to solve computational problems, creating and testing the necessary routines.
- understand how to analyse and evaluate the accuracy of a computational routine.
- develop skills in basic programming, code organisation
- understand at an introductory level key concepts in modern scientific programming - classes, parallel techniques

In-Lab Assessment

Due: **Weekly**
Weighting: **25%**

Laboratory exercises (computer laboratory; based on structured class work sheets and note books)

On successful completion you will be able to:

- This unit is designed to give the students a tool set of computational techniques that will be useful in their research and future careers. Students will be able to:
- demonstrate an understanding of the methods covered.
- apply their knowledge to solve computational problems, creating and testing the necessary routines.
- understand how to analyse and evaluate the accuracy of a computational routine.
- develop skills in basic programming, code organisation
- understand at an introductory level key concepts in modern scientific programming - classes, parallel techniques

Project

Due: **Week 12**
Weighting: **25%**

Individual projects - 4 weeks at the end of the semester with close supervision and a final report/ in class presentation

On successful completion you will be able to:

- This unit is designed to give the students a tool set of computational techniques that will be useful in their research and future careers. Students will be able to:
- understand the framework of techniques and which techniques would be most applicable to a given problem.
- demonstrate an understanding of the methods covered.
- apply their knowledge to solve computational problems, creating and testing the necessary routines.
- understand how to analyse and evaluate the accuracy of a computational routine.
- develop skills in basic programming, code organisation

Final Examination

Due: **Exam Weeks**
Weighting: **25%**

Final exam based on a combination of written and oral examination

On successful completion you will be able to:

- This unit is designed to give the students a tool set of computational techniques that will be useful in their research and future careers. Students will be able to:

- understand the framework of techniques and which techniques would be most applicable to a given problem.
- demonstrate an understanding of the methods covered.
- apply their knowledge to solve computational problems, creating and testing the necessary routines.
- understand how to analyse and evaluate the accuracy of a computational routine.
- develop skills in basic programming, code organisation

Delivery and Resources

- New and Improved Unit Description available HERE.
- Textbook requires is called "Introduction to Computational Science", Angela B. Shiflet and George W. Shiflet and is available in the on-campus bookstore
- All the material will be available via ilearn
- Course will use Matlab and Vensim (Systems Dynamics simulation: free for educational use)

Unit Schedule

Monday 16:00-18:00 Location EMC G240

Tuesday: 14:00-17:00 Location: EMC G210

All contact hours are a combination of lecture and interactive labs.

The course will start in Week w of S1 for the Labs to be setup.

Policies and Procedures

Macquarie University policies and procedures are accessible from Policy Central. Students should be aware of the following policies in particular with regard to Learning and Teaching:

Academic Honesty Policy http://mq.edu.au/policy/docs/academic_honesty/policy.html

Assessment Policy http://mq.edu.au/policy/docs/assessment/policy.html

Grading Policy http://mq.edu.au/policy/docs/grading/policy.html

Grade Appeal Policy http://mq.edu.au/policy/docs/gradeappeal/policy.html

Grievance Management Policy http://mq.edu.au/policy/docs/grievance_management/policy.html

Disruption to Studies Policy http://www.mq.edu.au/policy/docs/disruption_studies/policy.html *The Disruption to Studies Policy is effective from March 3 2014 and replaces the Special Consideration Policy.*

In addition, a number of other policies can be found in the Learning and Teaching Category of Policy Central.

Student Code of Conduct

Macquarie University students have a responsibility to be familiar with the Student Code of Conduct: https://students.mq.edu.au/support/student_conduct/

Results

Results shown in *iLearn*, or released directly by your Unit Convenor, are not confirmed as they are subject to final approval by the University. Once approved, final results will be sent to your student email address and will be made available in eStudent. For more information visit ask.mq.edu.au.

Student Support

Macquarie University provides a range of support services for students. For details, visit http://students.mq.edu.au/support/

Learning Skills

Learning Skills (mq.edu.au/learningskills) provides academic writing resources and study strategies to improve your marks and take control of your study.

- Workshops
- StudyWise
- Academic Integrity Module for Students
- Ask a Learning Adviser

Student Services and Support

Students with a disability are encouraged to contact the Disability Service who can provide appropriate help with any issues that arise during their studies.

Student Enquiries

For all student enquiries, visit Student Connect at ask.mq.edu.au

IT Help

For help with University computer systems and technology, visit http://informatics.mq.edu.au/help/.

When using the University's IT, you must adhere to the Acceptable Use Policy. The policy applies to all who connect to the MQ network including students.

Graduate Capabilities

PG - Capable of Professional and Personal Judgment and Initiative

Our postgraduates will demonstrate a high standard of discernment and common sense in their professional and personal judgment. They will have the ability to make informed choices and decisions that reflect both the nature of their professional work and their personal perspectives.

This graduate capability is supported by:

Learning outcomes

- This unit is designed to give the students a tool set of computational techniques that will be useful in their research and future careers. Students will be able to:
- understand at an introductory level key concepts in modern scientific programming - classes, parallel techniques

PG - Discipline Knowledge and Skills

Our postgraduates will be able to demonstrate a significantly enhanced depth and breadth of knowledge, scholarly understanding, and specific subject content knowledge in their chosen fields.

This graduate capability is supported by:

Learning outcomes

- This unit is designed to give the students a tool set of computational techniques that will be useful in their research and future careers. Students will be able to:
- understand the framework of techniques and which techniques would be most applicable to a given problem.
- demonstrate an understanding of the methods covered.
- apply their knowledge to solve computational problems, creating and testing the necessary routines.
- understand how to analyse and evaluate the accuracy of a computational routine.
- develop skills in basic programming, code organisation
- understand at an introductory level key concepts in modern scientific programming - classes, parallel techniques

PG - Critical, Analytical and Integrative Thinking

Our postgraduates will be capable of utilising and reflecting on prior knowledge and experience, of applying higher level critical thinking skills, and of integrating and synthesising learning and knowledge from a range of sources and environments. A characteristic of this form of thinking is the generation of new, professionally oriented knowledge through personal or group-based

critique of practice and theory.

This graduate capability is supported by:

Learning outcomes

- This unit is designed to give the students a tool set of computational techniques that will be useful in their research and future careers. Students will be able to:
- understand the framework of techniques and which techniques would be most applicable to a given problem.
- demonstrate an understanding of the methods covered.
- apply their knowledge to solve computational problems, creating and testing the necessary routines.
- understand how to analyse and evaluate the accuracy of a computational routine.
- develop skills in basic programming, code organisation
- understand at an introductory level key concepts in modern scientific programming - classes, parallel techniques

PG - Research and Problem Solving Capability

Our postgraduates will be capable of systematic enquiry; able to use research skills to create new knowledge that can be applied to real world issues, or contribute to a field of study or practice to enhance society. They will be capable of creative questioning, problem finding and problem solving.

This graduate capability is supported by:

Learning outcomes

- This unit is designed to give the students a tool set of computational techniques that will be useful in their research and future careers. Students will be able to:
- understand the framework of techniques and which techniques would be most applicable to a given problem.
- demonstrate an understanding of the methods covered.
- apply their knowledge to solve computational problems, creating and testing the necessary routines.
- understand how to analyse and evaluate the accuracy of a computational routine.
- develop skills in basic programming, code organisation
- understand at an introductory level key concepts in modern scientific programming - classes, parallel techniques

PG - Effective Communication

Our postgraduates will be able to communicate effectively and convey their views to different

social, cultural, and professional audiences. They will be able to use a variety of technologically supported media to communicate with empathy using a range of written, spoken or visual formats.

This graduate capability is supported by:

Learning outcomes

- This unit is designed to give the students a tool set of computational techniques that will be useful in their research and future careers. Students will be able to:
- apply their knowledge to solve computational problems, creating and testing the necessary routines.
- understand at an introductory level key concepts in modern scientific programming - classes, parallel techniques

PG - Engaged and Responsible, Active and Ethical Citizens

Our postgraduates will be ethically aware and capable of confident transformative action in relation to their professional responsibilities and the wider community. They will have a sense of connectedness with others and country and have a sense of mutual obligation. They will be able to appreciate the impact of their professional roles for social justice and inclusion related to national and global issues

This graduate capability is supported by:

Learning outcomes

- This unit is designed to give the students a tool set of computational techniques that will be useful in their research and future careers. Students will be able to:
- apply their knowledge to solve computational problems, creating and testing the necessary routines.
- understand at an introductory level key concepts in modern scientific programming - classes, parallel techniques