Imperial College London

Basic details

UID		Cohorts covered	Earliest cohort: 2023-24; Latest cohort:
Long title	Computational Physics		
New code	PHYS60012	New short title	

Brief description of module *(approx. 600 chars.)*

Computational Physics is about the application of computational methods to solve problems in physics. In core courses, you will have seen how physical systems can be described mathematically, often using a differential equation. Many of the examples encountered so far, in mechanics, electromagnetism and quantum physics, have analytic solutions. In real life, the number of such problems is limited and numerical methods are used to solve most problems in mathematical physics, even for apparently simple systems. In this course, you will learn how to select and apply various techniques to solve mathematical physics problems, as well as how to evaluate the suitability of numerical methods. It will give you the basic understanding to find numerical solutions of problems encountered in physics at research level. The skills acquired will be relevant for future work in both theoretical and experimental physics as well as mathematical modelling. The course does not teaching coding and students taking the course should have a significant amount of previous coding experience.

1080 characters

Available as a standalone module/ short course? N

Statutory details

	ECTS	CATS	Non-credit
Credit value	7.5	15	N

HECOS codes:

FHEQ level: Level 6

Allocation of study hours

	Hours	
Lectures	16	
Group teaching	0	*Incl. seminars, tutorials, problem classes.*
Lab/ practical	30	
Other scheduled	7	*Incl. project supervision, fieldwork, external visits.*
Independent study	134.5	*Incl. wider reading/ practice, follow-up work, completion of assessments, revi*
Placement	0	*Incl. work-based learning and study that occurs overseas.*
Total hours	187.5	
ECTS ratio	25.00	

Project/placement activity

Is placement activity allowed?	No

Module delivery

Delivery mode	Taught/ Campus	Other	
Delivery term		Other	Term 1

Ownership

Primary department	Physics
Additional teaching department	None
Delivery campus	South Kensington

Collaborative delivery

Collaborative delivery?	N
External institution	N/A
External department	N/A
External campus	N/A

Associated staff

Role	CID	Given name	Surname
Module leader		Mark	Scott
Module leader		Paul	Dauncey

Learning and teaching

Module description

Learning outcomes	On completion of this module you will be able to: - Identify fundamental problem types in computational physics (root-finding, interpolation, matrix inversion, optimisation, integration, differential equations) - Understand the implementation of bisection for root-finding, cubic splines for interpolation, and assorted basic methods for solving matrix equations. - Select suitable random number generators and use them in 'Monte Carlo' methods for multi-dimensional function minimisation and integration. - Select, assess (in terms of accuracy, stability & efficiency) and understand the implementation of finite-difference methods to perform numerical integration and solve ordinary and partial differential equations in physics. - Understand how to solve physics problems using combinations of any of the above techniques. - Understand when numerical library routines can be reliably used to solve problems.
Module content	- IEEE variable types and floating-point arithmetic - Root-finding - Basic matrix-inversion methods - Interpolation - Random numbers: How to generate non-uniform random distributions. Using them to efficiently calculate multi-dimensional integrals. - Optimisation problems: Newton and Monte Carlo methods for finding the minimum of general multi-dimensional functions. - Fourier-transform methods - Analysis of the accuracy and Contentstability of numerical methods for solving differential equations. - Numerical integration via finite-difference methods. - Solution of initial-value ODE problems (Runge-Kutta and related finite-difference methods) - Solution of boundary-value ODEs (shooting and related methods) - Solution of initial-value parabolic and hyperbolic PDEs using finite difference methods. - Use of matrix methods to solve elliptic (boundary-value) differential equations.LearningandTeachingStrategy
Learning and Teaching Approach	Students will be taught over one term using a combination of lectures, practical sessions with demonstrator assistance, office hours and non-assessed coursework.
Assessment Strategy	Written exam 50% Project 50%
Feedback	Problem sheets with worked solutions will be provided for all material. Detailed written feedback and preliminary marks will be given on submitted project reports. Verbal feedback and advice will be available in the practical sessions, during office hours and during the lectures themselves.
Reading list	Lecture notes will be provided and no additional books are required to be purchased by the students. Further discussion of material covered by the course can be found in: - Press et al. Numerical recipes: the art of scientific computing. 3rd Ed. Cambridge University Press. - Hoffmann. Numerical methods for engineers and scientists. 2nd Ed. Marcel Dekker. - Gerald. Applied numerical analysis. 7th Ed. Pearson.

Quality assurance

Date of first approval	
Date of last revision	
Date of this approval	

Office use only

QA Lead	
Department staff	
Date of collection	
Date exported	

Module leader	Mark Scott	Date imported	
Notes/ comments			

Template version 16/06/2017

Programme structure

Associated modules

UID	Legacy code	Module title	Requisite type

UID	Legacy code	Module title	Requisite type

Assessment details

Grading method	Numeric	Pass mark	40%

Assessments

Assessment type	Assessment description	Weighting	Pass mark	Must pass?
Examination	One hour written exam	50%	40%	N
Practical	Project	50%	40%	N
		100%		