

SEE8222 Scientific Computing in Energy and Environmental Science

Course Title:	Scientific Computing in Energy and Environmental Science
Course Code:	SEE8222
Course Duration:	One semester
Credit Units:	3
Level:	R8
Medium of Instruction:	English
Prerequisites:	Nil
Precursors:	Students should have knowledge of computer programming equivalent to CS1302 and must be familiar with at least one high-level programming language (preferably Python). Should also be familiar with elementary partial differential equations (at the level of MA3512 or equivalent)
Equivalent Courses:	Nil
Exclusive Courses:	MA6612 Numerical Partial Differential Equations

Course Aims

This course aims to provide students with an understanding of numerical methods for energy and environment.

Course Intended Learning Outcomes (CILOs)

(State what the student is expected to be able to do at the end of the course according to a given standard of performance.)

Upon successful completion of this course, students should be able to:

No.	CILOs	Weighting (if applicable)
1.	Describe why numerical solutions are required for energy and environment problems	0.5
2.	Apply numerical methods for optimisation and for solving linear systems	2.0
3.	Apply numerical methods for solving ordinary differential equations	2.5
4.	Apply the finite-difference method for partial differential equations	2.5
5.	Apply the finite-element and finite-volume methods for partial differential equations	1.5
6.	Apply the spectral method for partial differential equations	1.0

Teaching and Learning Activities (TLAs)

(Indicative of likely activities and tasks designed to facilitate students' achievement of the CILOs. Final details will be provided to students in their first week of attendance in this course.)

CILO No.	Lectures (hours)	Tutorials (hours)	Total Hours
CILO 1	2	2	4
CILO 2	6	2	8
CILO 3	5	2	7
CILO 4	8	3	11
CILO 5	3	2	5
CILO 6	2	2	4
Total (hours)	26	13	39

Lectures will cover basic theory. Tutorials will discuss implementation in Python and introduce students to scientific programming techniques.

Assessment Tasks/Activities

(Indicative of likely activities and tasks designed to assess how well the students achieve the CILOs. Final details will be provided to students in their first week of attendance in this course.)

CILO No.	Computational Assignments	Mini Project	Final	Total
CILO 1	0	0	4	4
CILO 2	13	0	4	17
CILO 3	12	10	3	25
CILO 4	8	15	3	26
CILO 5	0	15	3	18
CILO 6	7	0	3	10
Total	40	40	20	100

Duration of exam: 120 minutes for final

Mark allocation: 80% by course work, 20% by exams

To pass the course, a student must do ALL of the following:

- 1) obtain at least 30% of the total marks allocated for the course work (computational assignments, mini project);
- 2) obtain at least 30% of the total marks allocated for the final examination;
- 3) meet the criteria listed in the section on Grading of Student Achievement.

Grading of Student Achievement: Refer to Grading of Courses in the Academic Regulations for Postgraduate Degrees (Attachment) and to the Explanatory Notes.

Grade A

The student has a thorough understanding of the numerical techniques, including knowledge of assumptions, appropriateness and computational efficiency. The student is also able to demonstrate mastery of the theory by solving computational problems related to energy and environment.

Grade B

The student has a good understanding of the numerical techniques. The student can apply the theory to computational problems related to energy and environment, though without great fluency or full knowledge of the inherent limitations.

Grade C

The student has a good working knowledge of the numerical techniques. The student has some ability to apply the theory to computational problems related to energy and environment, but is generally incapable of producing complete solutions.

Grade D

The student has some knowledge of the numerical techniques. The student can produce a solution to computational problems related to energy and environment if significant assistance is provided.

Grade F

The student has little understanding of the numerical techniques. The student generally cannot produce solutions to computational problems related to energy and environment even if significant assistance is provided, and/or the student's work shows evidence of plagiarism.

Part III

Keyword Syllabus

- CILO1: numerical method, algorithm, types of differential equations, error, convergence
- CILO2: numerical linear algebra, optimisation, Monte Carlo method, iterative methods
- CILO3: time step, accuracy, adaptive methods, explicit and implicit schemes, Euler and Runge-Kutta methods, stiff equations, geometric integration
- CILO4: finite-difference stencil, linear stability analysis, CFL condition, Lagrangian and semi-Lagrangian methods, conservation laws
- CILO5: basis and test functions, variational formulation, Galerkin approximation
- CILO6: direct spectral methods, pseudospectral method, Fast Fourier transform, exponential convergence

Recommended Reading

Text(s)

D. R. Durran, *Numerical Methods for Fluid Dynamics*, Springer, Second Edition, 2010.

(Available

from http://encore.lib.cityu.edu.hk/iii/encore/record/C__Rb3991401__Sdurran+numerical__Origtr esult__X2?lang=eng&suite=pearl)

W.H. Press et al, *Numerical Recipes: the art of scientific computing*, Cambridge University Press, Third Edition, 2007.

R.H. Pletcher, J.C. Tannehill, D.A. Anderson, *Computational fluid mechanics and heat transfer*, CRC Press, 2013.

L.N. Trefethen, *Spectral methods in MATLAB*, SIAM, 2000.

H.P. Langtangen, *A primer on scientific programming with Python*, Springer, 2012.

(Available from <http://libweb.cityu.edu.hk/cgi-bin/er/db/splin3bk.pl?978-3-642-30293-0>)

Online Resources

R.W. Riddaway and M. Hortal, “Numerical Methods”,

http://www.ecmwf.int/newsevents/training/lecture_notes/pdf_files/NUMERIC/Num_meth.pdf

ECMWF training course on "Numerical Methods and Adiabatic Formulation of

Models", http://www.ecmwf.int/newsevents/training/meteorological_presentations/2013/NM2013/index.html

“Tentative NumPy tutorial”, http://wiki.scipy.org/Tentative_NumPy_Tutorial

“The Python Tutorial”, <http://docs.python.org/2/tutorial/>

“Matplotlib Tutorial”, <http://www.loria.fr/~rougier/teaching/matplotlib/>