YACHAY TECH UNIVERSITY

SYLLABUS

1. General Information

A.	**School**	School of Physical Sciences and Nanotechnology	B.	**Major**	Nanotechnology
C.	**Course**	Computational Physics I	D.	**Code**	ECFN-NAN-1117
E.	**Level**	Not Applicable	F.	**Academic Term**	Second Semester 2025
G.	**Curricular Unit**	Professional	H.	**Study Mode**	In person
I.	**Total Hours**	144	J.	**Professors**	Wladimir Eduardo Banda Barragán
K.	**Weekly Class Schedule**	09:00 - 11:00 Monday 11:00 - 13:00 Tuesday 07:00 - 09:00 Thursday	L.	**Weekly Tutoring Schedule**	16:00 - 17:00 Tuesday 09:00 - 10:00 Thursday

2. Prerequisites and Corequisites

Prerequisites		Corequisites	
Course	**Code**	**Course**	**Code**

3. Course Description

This course provides an introduction to basic methods and techniques used in computational physics as well as an overview of recent progress made in several areas of scientific computing. The course describes basic concepts of object-oriented programming and includes detailed step-by-step examples of how to optimally utilise computers and programming languages to solve problems in physics. Topics range from data analysis, approximation and optimisation of functions, through numerical calculus and differential equations, to matrix operations and spectral analysis. Each section of the course includes practical examples on different areas of science and technology in which computational physics has played a major role in the last decade.

4. Learning Outcomes of the Course

- Develop object-oriented programming skills for scientific computing within Linux environments.
- Design algorithms and implement software (in Python) dedicated to data analysis and visualisation.
- Apply numerical methods and computational techniques to model physical systems, analyse images, and carry out Fourier analysis.
- Use computational methods for research applications on astrophysics, electromagnetism, particle physics, classical mechanics, quantum mechanics, and other areas of physics.

5. Units / Contents / Hours / Evaluation Instruments and Schedule

Curricular Units	Contents	Teaching Hours	Hours of Internship and Experimental Learning		Hours of Independent Learning	Evaluation Instruments and Schedule
			Hours in Contact	Hours not in Contact		
Unit 1. Data Analysis and Visualization	Introduction to computer science, Linux operating systems, and scientific programming	2	2	0	2	In-class quizzes and homework assignments with dates shared in a timely manner by the lecturer. Midterm exam (September 29 to October 7). Final exam (December 5 to 10).
	Errors and uncertainties in computing, computer algorithms, and programming languages	2	2	0	2	
	Object-oriented programming, data reading/writing, graphics, statistics, data fitting, and regression	2	2	0	2	
	Function approximation, interpolation and extrapolation, Spline approximation	2	2	0	2	
Unit 2. Linear algebra, matrices, and images in physics	Systems of linear and nonlinear equations. Gaussian and iterative methods. Eigenvalue problems	2	2	0	2	In-class quizzes and homework assignments with dates shared in a timely manner by the lecturer. Midterm exam (September 29 to October 7). Final exam (December 5 to 10).
	Array programming. Linear algebra, vectors, matrices, and images	2	2	0	2	
	Matrix operations, basic image processing, and visualisation tools	2	2	0	2	
	Scientific data formats, animations, and scientific visualisation	2	2	0	2	

Unit 3. Numerical Calculus and Optimization	Numerical differentiation, symbolic algebra, discretisation, and numerical error analysis	3	3	0	3	In-class quizzes and homework assignments with dates shared in a timely manner by the lecturer. Midterm exam (September 29 to October 7). Final exam (December 5 to 10).
	Numerical integration, symbolic algebra, discretisation, and numerical error analysis	3	3	0	3	
	Derivatives and integrals in N-dimensional data, based on computational grids. Applications for image analysis and data cubes	3	3	0	3	
	Numerical optimisation, root finding, and extreme values of functions	3	3	0	3	
Unit 4. Fourier Analysis and Applications	Fourier Series and introduction to Fourier Analysis	2	2	0	2	In-class quizzes and homework assignments with dates shared in a timely manner by the lecturer. Midterm exam (September 29 to October 7). Final exam (December 5 to 10).
	Discrete Fourier Transform and the Fast Fourier Transform Algorithm. Applications to 1D and 2D signals	2	2	0	2	
	Fourier Filter Design and Noise Removal in N-Dimensional Signals	2	2	0	2	
	Gaussian Smoothing and Edge Detectors. Applications to Scientific Imaging	2	2	0	2	
Unit 5. Special Topics in Computational Physics	Wavelet analysis and discrete wavelet transform. Applications to non-static signals	3	3	0	3	In-class quizzes and homework assignments with dates shared in a timely manner by the lecturer. Midterm exam (September 29 to October 7). Final exam (December 5 to 10).
	Introduction to Monte Carlo methods. Monte Carlo simulations	3	3	0	3	
	Applications of Monte Carlo, random walks, and radioactive decay	3	3	0	3	
	Ordinary differential equations and initial value problems	3	3	0	3	
	Total	48	48	0	48	144

6. Teaching Methodology

Learning Environments

Learning Environment	Characteristics
Real: Digital or printed materials designed by the lecturer.	**Real:** Classroom equipped with a white board, which promotes the development of communication skills through interactive in-person activities. Access to a laboratory with computers and specific software for programming, data analysis, and simulation activities.
Virtual: MOODLE platform, GitHub, the Internet, websites, theoretical classes via ZOOM or Google Meet, seminars/workshops, tutoring, and other technological resources.	**Virtual:** Educational platform for sharing presentations, homework, code, assessments, and student participation forums.
Classroom: Theoretical classes, seminars/workshops, exercise classes, tutoring.	

Learning Tools

Learning Tools	Characteristics
Interactive lectures include programming theory and assignments, in-class/lab assignments include programming exercises and quizzes on reading material, and individual and group projects include programming and research assignments.	These tools help students develop programming skills for designing and implementing software for applications in the physical sciences.

7. Information Sources (Bibliography)

7.1 Main

Author/s	Title of Work	Edition	Year of Publication	Publishing house - Country
Landau, Rubin	Computational physics : problem solving with python	3rd	2015	Wiley-VCH; John Wiley - Germany
Kong, Qingkai; Siauw, Timmy; Bayen, Alexandre	Python Programming And Numerical Methods: A Guide For Engineers And Scientists	1st	2020	Elsevier: https://pythonnumericalmethods.berkeley.edu/notebooks/Index.html

7.2 Complementary				
Author/s	**Title of Work**	**Edition**	**Year of Publication**	**Publishing house - Country**
Pang, Tao	An introduction to computational physics	2nd	2006	Cambridge University Press – United States

8. Student's Evaluation

8.1 First Term of the Period*

Formative Evaluation		Laboratory**		Midterm Evaluation		Total
Homework average (code routines and project reports)	10%	Classwork average (reading quizzes and programming exercises)	10%	1 Midterm Exam	30%	50 %
Subtotal	**10%**	**Subtotal**	**10%**	**Subtotal**	**30%**	

8.2 Second Term of the Period

Formative Evaluation		Laboratory**		Final Evaluation		Total
Homework average (code routines and project reports)	10%	Classwork average (reading quizzes and programming exercises)	10%	1 Final Exam	30%	50 %
Subtotal	**10%**	**Subtotal**	**10%**	**Subtotal**	**30%**	

Evaluation Considerations

Make Up Exam

According to Article 61 of the UITEY Academic Regulations, "if students have a final grade point average of 4.8 to 5.9 in the subject, they will be entitled to a make-up exam at the end of the term. The content of this exam must reflect all aspects covered in the subject during the academic term. (…).

A passing grade on a make-up exam will raise the student's total grade to 6.0 (the minimum passing grade)."

Student Attendance to Academic Activities

Students must comply with Articles 82 and 83 of the UITEY Academic Regulations, which state:
*"**Article 82.- Student Attendance at Academic Activities**. – Students must attend all academic activities, both synchronous and asynchronous, such as classes, assessments, laboratory practices, and other activities that contribute to their overall education, on the established dates and times. Academic staff must ensure compliance with the attendance criteria and the learning components in contact with the faculty and experimental practice through attendance records in accordance with the mechanisms defined by the Academic Vice-Rectorate.*

For second language learning subjects, a minimum attendance of eighty percent (80%) is required for passing. The minimum attendance required for passing the other subjects will be seventy percent (70%).

Article 83.- Justification for absence. *- Students who are absent must present supporting documents within a maximum period of five (5) days, counting from the date of absence.* *The justifications will be validated by the teacher or the competent unit. Once the justification has been approved, the teacher, in his or her discretion, may define the criteria for making up activities missed due to non-attendance."*

(*) Teaching staff will record the scores generated up to the first term of the period in Moodle, by the deadline established in the academic calendar. To determine the weights of each item, the provisions of Article 63 of the UITEY Academic Regulations must be observed.
(**) For courses without a laboratory component, indicate: N/A.

9. General Considerations

- Students are responsible for ensuring the academic integrity of their submitted assignments and exams.
- Cheating in exams, plagiarising, and copying solutions from other students, from chatbots automated by artificial intelligence, from solution manuals, or from previous years' solutions are all breaches of academic integrity.
- Academic misconduct will be penalised according to the University's regulations.
- Assignment deadlines and exam dates will be discussed and agreed upon in class. Once fixed, they are hard deadlines.

Prepared by	Reviewed by	Approved by
Professor - Professors	**Designated Personnel**	**Dean – Language Director**
Signature and Date: Firmado electrónicamente por: WLADIMIR EDUARDO BANDA BARRAGAN Validar únicamente con FirmaEC	**Signature and Date:** Firmado electrónicamente por: CLARA INES ROJAS CELY Validar únicamente con FirmaEC	**Signature and Date:**