

## Programación

---

- Número de créditos: 10
- Semestre: 1
- Horas a la semana: (presenciales y de trabajo al estudiante)
- Teoría: 4
- Práctica: 2
- Autoestudio:
- Requisitos: NA
- Clave:
- Asignatura: Etapa Básica
- Materia asociada a la Línea de investigación:

---

Descripción de la asignatura: El objetivo del curso es que el estudiante aprenda el manejo de python para aplicaciones científicas con el uso de los módulos numpy, scipy, pandas entre otros. Programación en paralelo y cuda en python.

Contenido: Introducción a python, numpy, Matplotlib, scipy, sympy, pandas

Índice temático:

- **Numpy ( 16 sesiones)**

Declaración de arreglos y matrices, propiedades de arreglos, operaciones entre arreglos, funciones universales, slicing, arreglos multidimensionales, aplicaciones.

- **SciPy y métodos numéricos ( 8 sesiones)**

integración, interpolación, raíces de polinomios, ODEs, álgebra lineal, FFTs, ajustes

- **SymPy (4 sesiones)**

crear expresión, factorizar un polinomio, raíces, integración y diferenciación, convertir una expression ascii a sympy, definición de unidades.

- **Programación en Paralelo con Python (16 sesiones)**

Contextualizando programación en paralelo, concurrente y distribuida; diseño de algoritmos en problema; identificar problemas paralelizables usando módulos para hilos y características concurrentes; multiprocesos.

- **Introducción a Fortran (12 sesiones)**

Declaración, tipo de variables, ciclos, condicionales, lectura y escritura de variables, funciones y subrutinas.

- **Programación en Paralelo en Fortran ( 8 sesiones)**

**Bibliografía:**

- Clua Führer, Jan Erik Solem, Iliver Verdier, *Scientific Computing with Python 3* (2016), Packt Publishing
- Stephen J. Chapman, *FORTTRAN for Scientist & Engineer*, McGraw-Hill Education; Edición: 4th ed. (2017)
- John V. Guttag, *Introduction to Computation and Programming Using Python with Application to Understanding Data* ( 2016 ), MIT Press.
- Hans Petter Langtangen, *A Primer on Scientific Programming with Python* (2014), Springer
- Ivan Idris, *NumPy Beginner's Guide, Second Edition* (2013). Packt Publishing.
- Allen Chi Shing Yu, Claire Yik Lok Chung, Aldrin Kay Yuen Yim, *Matplotlib 2.x by examples* (2017), Packt Publishing.
- Ian Chivers and Jane Sleightholme, *Introduction to Programming with FORTRAN*, Springer; Edición: 4th 2018 ed. (2018).

**Planeación Educativa**

**Competencias a Desarrollar:**

El estudiante comprenderá el uso de los lenguajes de programación PYTHON y FORTRAN, de una manera estructurada y eficiente.

El profesor promoverá el trabajo fuera del aula, mediante tareas las cuales tendrán que ser escritas para posteriormente revisarlas en la sesión de talleres. De esta forma los alumnos desarrollaran su capacidad de comunicar sus ideas de forma escrita y oral.

Mediante la sesión de talleres, el alumno aprenderá a discutir con sus compañeros sus ideas y diferentes soluciones para un problema, lo cual fomentará el trabajo en equipo, el respeto y la tolerancia.

**Generales:**

## Unidad Académica de Ciencia y Tecnología de la Luz y la Materia

[lumat@uaz.edu.mx](mailto:lumat@uaz.edu.mx)

El estudiante desarrollará capacidades de abstracción, análisis y síntesis, organización y planificación del tiempo, habilidad para trabajar de forma autónoma y al mismo tiempo el trabajo en equipo, habilidad para transmitir sus conocimientos de forma oral y escrita, creatividad, identificación y

planteamiento de problemas, motivar y conducir hacia metas comunes, respeto por la diversidad y multiculturalidad, compromiso con la calidad.

### *Específicas:*

Dominio de los conceptos básicos de la computación científica a través del uso de diferente software científico. Capacidad para expresarse correctamente. Capacidad de abstracción incluido el desarrollo lógico. Capacidad para comprender problemas. Capacidad para presentar sus razonamientos de forma apropiada para sus pares académicos, tanto oralmente como por escrito. Capacidad para detectar inconsistencias.

<b>Matriz Educacional</b>			
Resultados del Aprendizaje	Actividades Educativas	Horas	Evaluación
		Teóricas, Prácticas, autoestudio, y numérico-computacionales	
1. Numpy	Asistencia a clases, tareas a casa, clases prácticas	Teóricas 2 Prácticas 4 Autoestudio 4	Proyecto y examen
2. Scipy y métodos numéricos	Asistencia a clases, tareas a casa, clases prácticas	Teóricas 2 Prácticas 4 Autoestudio 4	Proyecto y examen
3. SymPy	Asistencia a clases, tareas a casa, clases prácticas	Teóricas 2 Prácticas 4 Autoestudio 4	Proyecto
4. Programación en paralelo en PYTHON	Asistencia a clases, tareas a casa, clases prácticas	Teóricas 2 Prácticas 4 Autoestudio 4	Proyecto y examen
5. Introducción a FORTRAN	Asistencia a clases, tareas a casa, clases prácticas	Teóricas 2 Prácticas 4 Autoestudio 4	Proyecto y examen
6. Programación en paralelo en FORTRAN	Asistencia a clases, tareas a casa, clases prácticas	Teóricas 2 Prácticas 4 Autoestudio 4	Proyecto y examen