

<b>PROGRAMA - AÑO 2024</b>	
<b>Espacio Curricular:</b>	Programación en Python
<b>Carácter:</b>	Electivo <b>Período:</b> 1° Semestre
<b>Carrera/s:</b>	Licenciaturas en Ciencias Básicas con Orientación en Biología, Física, Matemática o Química Licenciatura en Geología. Profesorado de Grado Universitario en Ciencias Básicas con orientación en Física, Química, Matemática o Biología
<b>Profesor Responsable:</b>	Emmanuel N. Millán
<b>Equipo Docente:</b>	
<b>Carga Horaria:</b> 60 hrs; 6 horas semanales, 2 horas de teoría y 4 de práctica.	
<b>Recomendaciones de Cursado:</b>	

### 1-EXPECTATIVAS DE LOGRO

- Dominar los conceptos básicos del lenguaje Python.
- Identificar y organizar los datos pertinentes al problema, evaluando el contexto particular.
- Comprender las ventajas y desventajas del lenguaje de Python.
- Utilizar conceptos básicos de complejidad computacional en Python.
- Conocer los diversos paquetes científicos disponibles para Python para funcionalidad avanzada.
- Conocer las fuentes de información disponibles para el usuario para seguir mejorando sus habilidades.
- Resolver ejercicios prácticos sencillos y específicos utilizando códigos propios en lenguaje Python

### 2-DESCRIPTORES

El rol de Python en el procesamiento de datos. Estructuras de datos, controles de flujo, funciones, y paquetes. Análisis de datos con paquetes científicos especializados. Cuadernos Jupyter y programación interactiva. Visualización con aplicaciones científicas

### 3-CONTENIDOS ANALÍTICOS

#### Unidad 1: Introducción general

Introducción general a Python. Sintaxis del lenguaje. Lógica binaria. Lenguajes de programación. Introducción a los cuadernos Jupyter y la programación interactiva. Peculiaridades de Python. Control de paquetes.

#### Unidad 2: Resolución de problemas

Problemas y algoritmos. Conceptos, características y clasificación de algoritmos. Partes y

representación. Clasificación de problemas y etapas para su resolución. Estructuración de datos. Constantes y variables. Entrada, proceso y salida. Tipos de datos: numéricos, lógicos y de caracteres.

### **Unidad 3: Diagramación**

Pseudocódigo. Características de un lenguaje estructurado. Diagramas de Flujo. Elementos, reglas y usos. Sentencias secuenciales, condicionales y repetitivas. Modularización de algoritmos. Concepto de funciones y procedimientos.

### **Unidad 4: Estructuras de datos**

Arreglos, diccionarios, listas, y tuplas. Carga, búsqueda, y muestra de datos. Operaciones con estructuras de datos. Métodos de ordenamiento, búsqueda e intercalación.

### **Unidad 5: Entorno de desarrollo**

Estructura de un programa. Bibliotecas de funciones. Tipos de datos. Operadores. Funciones de entrada y salida de datos. Asignación. Estructuras de control de flujo del programa. Entrada y salida de archivos de texto, o archivos de datos.

### **Unidad 6: Funciones en Python**

Modularización de programas. Funciones: declaración, llamado y definición de funciones en Python. Paso de argumentos a una función. Variables. Alcance de las variables: locales y globales.

### **Unidad 7: Análisis de datos científicos - Aplicaciones**

Introducción y aplicación del ecosistema SciPy y sus varios paquetes para trabajo con datos científicos. Aplicación del ecosistema a ejemplos prácticos de conjuntos de datos empíricos en distintas áreas de la ciencia. Revisión de bibliotecas como Numpy, Pandas, scikit-image y scikit-learn.

### **Unidad 8: Visualización de datos**

Introducción al paquete Matplotlib. Trazar gráficos unidimensionales, y multidimensionales. Producción de gráficos de calidad profesional. Aplicación del paquete Matplotlib a ejemplos en diferentes áreas de la ciencia.

## **4-BIBLIOGRAFÍA**

1. C. Horstmann and R. Necaie, Python for everyone, Wiley, Hoboken, New Jersey, 1st edn., 2014.
2. Cay Horstmann, Rance D. Necaie. Python for Everyone. Wiley. 2014 ISBN 978-1-118-62613-9
3. Andres Marzal Varo, Isabel Gracia Luengo, Pedro Garcia Sevilla. Introducción a la programación con Python 3. Universitat Jaume. ISBN: 978-84-697-1178-1
4. Scipy-cookbook, <https://scipy-cookbook.readthedocs.io>. (accedido Febrero 2023)
5. Matplotlib, <https://matplotlib.org/index.html>. (accedido Febrero 2022)
6. NumPy Manual, <https://docs.scipy.org/doc/numpy/>. (accedido Febrero 2023)
7. D. Beazley and B. Jones, Python Cookbook, O'Reilly, Sebastopol California, 3rd edn., 2013.
8. J. VanderPlas, Python Data Science Handbook, O'Reilly Media, Sebastopol California, 1st edn., 2016.
9. W. McKinney, Python for data analysis, O'Reilly, Sebastopol California, 1st edn., 2012.
10. SymPy, <https://docs.sympy.org/latest/index.html>. (accedido Febrero 2023)
11. Pandas, <http://pandas.pydata.org/pandas-docs/stable/>. (accedido Febrero 2023)
12. S. Tosi, Matplotlib for Python developers, Packt Pub., Birmingham, U.K., 1st edn., 2009.

## **5-METODOLOGÍA DE ENSEÑANZA Y EVALUACIÓN DURANTE EL CURSADO**

El contenido de la materia será introducido mediante presentaciones orales a cargo del docente. Se podrán utilizar herramientas de comunicación virtual para realizar clases de teoría y/o práctica de forma no-presencial. Todos los materiales de teoría y práctica estarán disponibles en el aula virtual. Además se realizarán clases prácticas en la sala de computación, donde los estudiantes aplicarán los conocimientos adquiridos resolviendo guías de problemas. Se realizará evaluación continua a través de trabajos prácticos, uno por cada tema de teoría desarrollado. Finalmente se realizará una única evaluación integradora escrita de conocimientos y competencias de programación.

## 6- CONDICIONES DE REGULARIDAD TRAS EL CURSADO

Los asistentes serán evaluados de manera continua durante el desarrollo de los aspectos prácticos de las clases, y mediante una evaluación integradora al finalizar el cursado.

Para poder acceder a dicha evaluación integradora, el asistente deberá tener un mínimo de 60% de asistencia al curso y deberán presentar todos los trabajos prácticos realizados durante el cursado. La evaluación integradora tomará la forma de ejercicios computacionales, en los cuales los asistentes deberán crear o completar programas informáticos en Python para resolver problemas específicos. El acceso al material del curso e internet no se restringirá durante la evaluación, ya que refleja un entorno de trabajo realista. Los ejercicios enviados se evaluarán en función de su capacidad para resolver el problema, la eficiencia del código, la legibilidad del código y la calidad de salida (cuando corresponda)

Para obtener la regularidad del espacio curricular la nota de esa evaluación deberá ser mayor o igual a 6 (seis).

## 7- SISTEMA DE APROBACIÓN Y/O PROMOCIÓN DEL ESPACIO CURRICULAR

Para promocionar el espacio curricular se debe tener una nota global (nota de la evaluación integradora y prácticas) mayor o igual a 7 (siete), cumplir con la asistencia del 70%, y tener aprobados todos los trabajos prácticos.

Para regularizar el espacio curricular se debe tener una nota global (nota de la evaluación integradora y prácticas) mayor o igual a 6 (seis) y cumplir con la asistencia del 60%.

Modalidad de evaluación para estudiantes libres

- Presentar y aprobar todos los trabajos prácticos la semana anterior a la mesa de examen, deben comunicarse con el docente para solicitar los trabajos prácticos.
- Aprobar, con una nota de 6 (seis) o mayor, una evaluación práctica que abarca todo el material del curso.
- Aprobar, con una nota de 6 (seis) o mayor, una evaluación oral para determinar el nivel de comprensión del material del curso.

**PROMOCIONABLE**

SI	X	NO	
----	---	----	--

## 8- CRONOGRAMA DE ACTIVIDADES

Cronograma propuesto sujeto a modificaciones con aviso y conformidad de estudiantes:

Miércoles	27/03/2024	Teoría y práctica sobre Anaconda e introducción a la programación en Python
Miércoles	03/04/2024	Teoría y práctica sobre Funciones matemáticas y Condicionales IF
Miércoles	10/04/2024	Teoría y práctica sobre control de flujo bucles y Funciones
Miércoles	17/04/2024	Teoría y práctica sobre estructuras de datos y lectura/escritura de archivos

Miércoles	24/04/2024	Teoría y práctica sobre Numpy
Miércoles	01/05/2024	FERIADO
Miércoles	08/05/2024	Teoría y práctica sobre Matplotlib y sci_libs
Miércoles	15/05/2024	Teoría y práctica sobre Pandas y ejercicios STEM
Miércoles	22/05/2024	Práctica, revisión de TPs y consulta
Miércoles	29/05/2024	Práctica, revisión de TPs y consulta
Miércoles	05/06/2024	EVALUACIÓN
Viernes	07/06/2024	ENTREGA EVALUACIÓN
Miércoles	12/06/2024	Recuperatorio
Viernes	14/06/2024	CIERRE DE ACTAS NOTAS



Emmanuel Millan

**FIRMA Y ACLARACIÓN  
PROFESOR RESPONSABLE DEL ESPACIO CURRICULAR**