



## PYTHON DE PRINCIPIANTE A EXPERTO – NIVEL AVANZADO

### I. INFORMACIÓN GENERAL

**Nombre del Curso:** Python de Principiante a Experto – Nivel Avanzado

**Duración:** 18 horas

**Profesor:** Alejandro Castro

**Requisitos:**

- Python de Principiante a Experto – Nivel Intermedio
- Conocimientos básicos de sistemas operativos: Windows, Linux o Mac.
- Matemáticas de nivel secundario.

### II. DESCRIPCIÓN DEL CURSO

El presente curso aborda los principales métodos avanzados empleados para incrementar la velocidad de software desarrollado utilizando el lenguaje Python, partiendo de las mejoras de velocidad mediante el uso de Cython o Python/C API hasta las mejoras intrínsecas que vienen de la paralelización y la concurrencia, conocimientos que serán reforzados mediante su aplicación en la implementación de un proyecto final.

### III. SUMILLA

Curso con un enfoque práctico donde el alumno aplicará de forma incremental los conocimientos adquiridos sobre programación utilizando el lenguaje de programación Python, finalizando con implementación de un proyecto final. Los temas abarcarán desde el uso de Cython, el Python/C API, la concurrencia y la paralelización.

### IV. OBJETIVOS

El principal objetivo del curso es darle herramientas potentes al alumno para que se pueda desempeñar como programador de Python capaz de implementar y/o trabajar en proyectos de avanzada complejidad en dicho lenguaje, haciendo uso de técnicas modernas de mejora de velocidad y las herramientas de trabajo más utilizadas y requeridas en la actualidad.



## V. PROGRAMA ANALÍTICO

### UNIDAD 1: MEJORANDO LA VELOCIDAD (10h)

1. Evaluando scripts: profiling
2. Python/C API
3. Numpy/C API
4. Cython

### UNIDAD 2: CONCURRENCIA Y PARALELIZACIÓN (8h)

1. Concurrencia vs paralelización
2. La librería threading
3. Hilos: Definición y características
4. La librería multiprocessing
5. Procesos: Definición y características

## VI. METODOLOGÍA

La metodología que se seguirá en el presente curso será la siguiente: primero se desarrollará cada uno de los temas de manera teórica, con apoyo de herramientas digitales gráficas, luego se reforzarán estos conceptos mediante la implementación y la solución de retos relacionados a los temas de clase y los precedentes.

## VII. EVALUACIÓN

Fórmula para el cálculo de la nota final. Cada una de las 2 unidades tendrá tareas de programación que una vez finalizado el curso podrán juntarse y complementarse para desarrollar un proyecto final planteado por el docente:

$$NOTA FINAL = \frac{1}{2} \sum_{i=1}^2 E_i + P$$

NOTA FINAL: Promedio final del curso

$E_i$ : Nota correspondiente a la  $i$ -ésima tarea de programación. Se califican de 0 a 12.

$P$ : Nota correspondiente a la presentación final del proyecto. Se califica de 0 a 8.

El docente seleccionará el medio de recepción de las evaluaciones y tomará en cuenta la asistencia y participación si así lo desea.