

## PLANIFICACIÓN DEL CURSO

### I. ACTIVIDAD CURRICULAR Y CARGA HORARIA

Asignatura: Programación	Código: ING1302
Semestre de la Carrera: II semestre	
Carrera: Obligatorio para todas las carreras de Ingeniería Civil	
Escuela: Escuela de Ingeniería	
Docente(s): Carol Moraga	
Ayudante(s): Cristian Herrera	
Horario: Martes y Jueves 10:15-11:45. Ayudantía Lunes 16:15-17:45	

Créditos SCT:	6
Carga horaria semestral <sup>1</sup> :	180 horas
Carga horaria semanal:	10 horas

Tiempo de trabajo sincrónico semanal:	3 horas
Tiempo de trabajo asincrónico semanal:	7 horas

### II. RESULTADOS U OBJETIVOS DE APRENDIZAJE ESPERADOS ESTE SEMESTRE

1) Descomponer un problema y hacer abstracciones utilizando el razonamiento lógico y algorítmico.
2) Plantear la solución a los problemas resultantes de la descomposición: diseñar contratos, especificar el propósito del código, generar casos de prueba y programar la solución.
3) Detectar y corregir errores de programación.

<sup>1</sup> Considere que 1 crédito SCT equivale a 27 horas de trabajo total (presencial/sincrónico y autónomo/asincrónico) en el semestre.

### III. UNIDADES, CONTENIDOS Y ACTIVIDADES

UNIDAD 1: Fundamentos de la programación			
Semana	Contenidos	Laboratorio	Actividades
<b>1</b> 13 de agosto	Introducción a la programación Feriado		
<b>2</b> 20 de agosto	Hardware y lógica en la programación Estructuras básicas y diagramas de flujo		
<b>3</b> 26 de agosto	Ejercicios de estructuración Variables de tipo numérico y string	Laboratorio de diagramas de flujo	
<b>4</b> 2 de septiembre	Input y Bucles (While, For) Sentencia condicional, operador AND y OR	Laboratorio variables y bucles. Instalación de Spyder	<b>Tarea 1 disponible en Ucampus</b>
<b>5</b> 9 de septiembre	Listas, tuplas y diccionarios Librería Turtle	Laboratorio de input y bucles	<b>Entrega Tarea 1 Jueves 12 septiembre</b>

<b>UNIDAD 2: Estructura de Datos y Funciones</b>			
<b>Semana</b>	<b>Contenidos</b>	<b>Laboratorio</b>	<b>Actividades</b>
<b>6</b> 16 de septiembre	Semana de autocuidado y aprendizaje autónomo	<b>FERIADO</b>	
<b>7</b> 23 de septiembre	CC1 Retroalimentación CC1	Laboratorio de condición, AND, OR aplicando listas tuplas y diccionarios	<b>CC1</b> <b>Martes 24 de septiembre</b>
<b>8</b> 30 de septiembre	Ejercicios con Turtle Funciones I	Laboratorio de listas tuplas y diccionarios	<b>Tarea 2 disponible en Ucampus</b>
<b>9</b> 7 de octubre	Funciones II Recursividad	Laboratorio de Turtle y Funciones	<b>Entrega Tarea 2</b> <b>Viernes 11 de octubre</b>
<b>10</b> 14 de octubre	Repaso CC2 CC2	Ejercicios de repaso CC2	<b>CC2</b> <b>Jueves 17 de octubre</b>

**UNIDAD 3: Tópicos Avanzados**

<b>Semana</b>	<b>Contenidos</b>	<b>Laboratorio</b>	<b>Actividades</b>
<b>11</b> 21 de Octubre	I/O de archivos 1 I/O de archivos 2	Laboratorio de funciones y recursividad	
<b>12</b> 28 de Octubre	Funciones y Detecciones de errores Feriado	Laboratorio de I/O de Archivos	
<b>13</b> 4 de Noviembre	Introducción a Programación Orientada a Objetos (POO) Herencia y polimorfismo	Laboratorio de Funciones y detecciones de errores	
<b>14</b> 11 de Noviembre	Numpy I Numpy II	Laboratorio de POO	
<b>15</b> 18 de Noviembre	Repaso CC3 CC3	Laboratorio de Numpy	<b>CC3</b> <b>Jueves 21 de</b> <b>Noviembre</b>

#### IV. CONDICIONES Y POLÍTICAS DE EVALUACIÓN

Se evaluará el aprendizaje del contenido presentado, mediante:

- 3 Controles de Cátedra (CC)
- 2 Tareas

El promedio ponderado simple de los 3 CC conforman la Nota de Cátedra (NC). El promedio de Tareas conforman la Nota de Actividades Complementarias (NAC). La aprobación del curso se logra con  $NC \geq 4,0$  y  $NAC \geq 4,0$ . La nota final se calcula como  $0,6*NC + 0,4*NAC$ . **Este curso no considera la realización de Examen.**

Las Tareas consisten en la resolución de un problema utilizando lo aprendido en las Cátedras. Se considera 1 tarea para la casa y 1 tarea desarrollada durante la hora de laboratorio.

Estudiantes que se ausenten justificadamente a alguno de los CC tendrán la oportunidad de recuperarlo a través de un control recuperativo cuyo justificativo se tramita a través de la Dirección de Asuntos Estudiantiles. **Las notas de tareas no se recuperan.**

Un/a estudiante que cometa plagio obtendrá un **1,0** en la evaluación y el caso será informado a Escuela de Ingeniería.

#### V. BIBLIOGRAFÍA Y RECURSOS OBLIGATORIOS

- Farrell, Joyce (2013): Introducción a la Programación. Lógica y Diseño. Cengage, 7ª Edición. Libro digital: <https://elibro.net/es/lc/bibliouoh/titulos/93265>
- Trejos, Omar y Luis Muñoz (2021): Introducción a la Programación con Python. RA-MA. Libro digital: <https://elibro.net/es/ereader/bibliouoh/230298>
- Marzal, Andres, Isabel Gracias y Pedro García (2014): Introducción a la Programación con Python 3. Universitat Jaume I. Libro digital: <https://elibro.net/es/ereader/bibliouoh/51760>

## VI. BIBLIOGRAFÍA Y RECURSOS COMPLEMENTARIOS

- Juganary-Mathieu, Mihaela (2014): Introducción a la Programación. Patria. Libro digital: <https://elibro.net/es/lc/bibliouoh/titulos/39449>
- Aquino, Miguel y Fernando Aquino (2021): Aprende Programación de Computadoras. Bubok. Libro digital: <https://elibro.net/es/ereader/bibliouoh/260300>
- Fernández, María A. (): Introducción Práctica a la Programación con Python. Universidad de Alcalá. Libro digital: <https://elibro.net/es/ereader/bibliouoh/124259>
- Python Tutorial w3schools.com. <https://www.w3schools.com/python/>.
- Google Colab: <https://colab.research.google.com/>
- Code Wars: <https://www.codewars.com/>
- El Libro de Python: <https://ellibrodepython.com/>
- Python Tutor: <https://pythontutor.com/python-compiler.html#mode=edit>