



### 1. INFORMACIÓN DEL CURSO:

<b>Nombre:</b> Programación Aplicada		<b>Número de créditos:</b> 7		<b>Clave:</b> 17404	
<b>Departamento:</b> Ciencias Computacionales		<b>Horas teoría:</b> 34		<b>Horas práctica:</b> 34	
				<b>Total, de horas por cada Semestre:</b> 68	
<b>Tipo:</b> Curso - Taller		<b>Prerrequisitos:</b> Ninguna		<b>Nivel:</b> Área Formación Optativa Abierta.	

### 2. DESCRIPCIÓN

#### Objetivo General:

Estudiar los conceptos básicos de programación, haciendo especial énfasis en comprender el proceso de traducción del planteamiento de un problema a su formulación computacional para resolverlo. Utilizará el lenguaje de programación Python para resolver problemas computables.

#### Objetivos Particulares:

1. Aplicar los conceptos básicos de la programación estructurada en la solución de los diferentes problemas planteados durante el curso.
2. Aprende el concepto de arreglo, los diferentes tipos que existen, así como el uso apropiado de los mismos en la solución de problemas.
3. Resuelve problemas complejos descomponiéndolos en partes más manejables, utilizando el concepto de función, así como el uso de los diferentes tipos de parámetros.
4. Crea programas que incluyan el diseño y manipulación de datos estructurados utilizando registros.

#### Contenido temático sintético (que se abordará en el desarrollo del programa y su estructura conceptual)

##### UNIDAD TEMÁTICA 1: INTRODUCCIÓN

- 1.1 Conceptos básicos de programación estructurada
- 1.2 Algoritmo
- 1.3 Estructuras de control

##### UNIDAD TEMÁTICA 2: ARREGLOS

- 2.1 Definición
- 2.2 Tipos de arreglos

##### UNIDAD TEMÁTICA 3: MANEJO DE FUNCIONES

- 3.1 Definición
- 3.2 Funciones sin paso de parámetros
- 3.3 Funciones con parámetros por valor

##### UNIDAD TEMÁTICA 4: COLECCIONES PRE-FABRICADAS EN PYTHON (LISTAS, TUPLAS, CONJUNTOS Y DICCIONARIOS)

- 4.1 Definición
- 4.2 Operaciones con una lista, tupla, conjunto, diccionario.
- 4.3 Arreglos con elementos de tipo lista.
  - 4.3.1 Definición

## Competencias a desarrollar

Transversales	Genéricas	Profesionales
<p>Abstraer la realidad para crear algoritmos de programación realizando análisis y síntesis.</p> <p>Obtener la capacidad para desarrollar algoritmos de programación y Aplicar los conocimientos en la práctica.</p> <p>Organizar y planificar el tiempo para el desarrollo de algoritmos</p> <p>Habilidades en el uso de las tecnologías de la información para poder realizar reportes en documentos de office y saber guardarlos en formato PDF.</p>	<p>Expresa y comunica de manera pertinente en distintos contextos.</p> <p>Desarrolla innovaciones y propone soluciones a problemas a partir de metodologías, métodos y principios establecidos.</p> <p>Trabaja en forma colaborativa.</p> <p>Toma decisiones de forma lógica sobre casos de estudios que coadyuven en el trabajo en equipo.</p> <p>Propone soluciones originales.</p> <p>Aprende por iniciativa e interés propio a lo largo de la unidad de aprendizaje.</p> <p>Organiza y regula su aprendizaje propio y en grupo de manera efectiva, para resolver problemas de tipo computacional.</p> <p>Aplica sus conocimientos en el desarrollo de proyectos o estudios de caso.</p>	<p>Aplica conocimiento de ciencias de la computación, de tecnologías de la información, y de las organizaciones, para desarrollar soluciones informáticas en el área de cálculo matemático y temas de la Ingeniería Industrial o Mecánica Eléctrica.</p> <p>Aplica el enfoque sistémico en el análisis y resolución de problemas de cálculo matemático y temas de la Ingeniería Industrial o Mecánica Eléctrica.</p> <p>Aplica fundamentos matemáticos, principios algorítmicos y teorías de Ciencias de la Computación en la modelación y diseño de soluciones informáticas de cálculo matemático y temas de la Ingeniería Industrial o Mecánica Eléctrica.</p>

Saber (conocimientos)	Saber hacer (habilidades)	Saber ser (actitudes y valores)
<p>Esencia operativa de las computadoras.</p> <p>Nociones básicas del acto de programar computadoras.</p> <p>Elementos básicos de un lenguaje de programación.</p> <p>Construcción de algoritmos para solucionar problemas sencillos.</p> <p>Funcionamiento de un programa a partir de su código fuente.</p>	<p>Abstracción.</p> <p>Manejo de Razonamiento lógico.</p> <p>Razonamiento matemático.</p> <p>Análisis y síntesis de información.</p> <p>Destreza en el uso de la computadora.</p> <p>Lectura y comprensión de idioma inglés.</p>	<p>Es responsable de en las fechas de entrega.</p> <p>Colabora con sus compañeros para mejorar el trabajo en equipo.</p> <p>Autogestión.</p> <p>Honestidad en el desarrollo de las actividades académicas.</p> <p>Responsabilidad y compromiso en las actividades individuales y por equipo.</p> <p>Toma decisiones y se proyecta como líder.</p>

## Modalidades de enseñanza aprendizaje

Solicita a los estudiantes lecturas previas acerca de los conceptos básicos de la computación y de la programación estructurada.

Indica a los alumnos organizarse en grupos y plantea una serie de ejercicios donde se apliquen los conceptos básicos.

Solicita a los estudiantes lecturas previas acerca de las Estructuras de control selectivas.

Plantea una serie de ejercicios donde se apliquen las estructuras de control selectivas

Solicita a los estudiantes lecturas previas acerca de las Estructuras de control repetitivas.

Plantea una serie de ejercicios donde se apliquen las estructuras de control repetitivas

Solicita a los estudiantes lecturas previas acerca de los diferentes tipos de arreglos

Plantea una serie de ejercicios donde se apliquen arreglos

Solicita a los estudiantes lecturas previas acerca de Programación Modular (funciones)

Plantea una serie de ejercicios donde se apliquen funciones

Solicita a los estudiantes lecturas previas acerca del manejo de listas, tuplas, diccionarios y conjuntos con funciones definidas por el usuario.

Plantea una serie de ejercicios donde el alumno diseñe algoritmos aplicando listas y funciones para la solución de un problema.

### Modalidad de evaluación

Esta UA requiere de la presentación de reportes, además de los respectivos programas en Lenguaje C, los cuales deben cumplir con lo siguiente:

- Entrega en tiempo
- Los reportes con la solución de los problemas deberán incluir lo establecido para cada evidencia.
- Queda estrictamente prohibido el plagio.
- Los programas deben ser funcionales.

Reporte escrito que contenga las definiciones de los conceptos básicos indicando la fuente de información utilizada.	3.4%
Reportes que contengan la solución de los problemas que les fueron planteados.	6.9%
Reporte escrito que contenga las definiciones de las estructuras selectivas. indicando la fuente de información utilizada.	3.4%
Reportes que contengan la solución de los problemas que les fueron planteados	6.9%
Reporte escrito que contenga las definiciones de las estructuras repetitivas. indicando la fuente de información utilizada.	3.4%
Reportes que contengan la solución de los problemas que les fueron planteados	9.2%
Reporte escrito que contenga las definiciones de los diferentes tipos de arreglos, indicando la fuente de información utilizada.	2.3%
Reportes que contengan la solución de los problemas que les fueron planteados	10.4%
Examen parcial	10%
Solicita a los estudiantes lecturas previas acerca de Programación Modular (funciones)	3.4%
Plantea una serie de ejercicios donde se apliquen funciones	10.4%
Reporte escrito que contenga las características de los registros, indicando la fuente de información utilizada.	1.1%
Reportes que contengan la solución de los problemas que les fueron planteados	9.2%
Examen Parcial	10%
Producto final	10%

### Campo profesional

Ingeniería industrial, Ingeniería en Computación.

### 3. BIBLIOGRAFÍA.

Autor (Apellido, Nombre)	Año	Título	Editorial
Cervantes Villagómez Ofelia D. Báez López Davis Arízaga Silva Antonio Castillo Juárez Esteban	2018	Python con aplicaciones a las matemáticas, ingeniería y finanzas	ALFAOMEGA
Cuevas Álvarez Alberto		Python 3 Curso práctico	RA-MA

Formato basado en el Artículo 21 del Reglamento General de planes de estudios de la U.de G.