



PROGRAMA DE ASIGNATURA PROGRAMACIÓN

ANTECEDENTES GENERALES

Carrera	INGENIERÍA CIVIL INDUSTRIAL			
Nombre de la asignatura	PROGRAMACION			
Código de la asignatura				
Año/Semestre	2023/2			
Coordinador Académico	Víctor Devia Jerez			
Equipo docente	Víctor Devia Jerez – Jacob Chaparro Soto- Eduardo Pérez Lobato – Gonzalo Flores – Andrés Alvear – Gustavo Osses			
Área de formación	PROFESIONAL			
Créditos SCT	3			
Horas de dedicación	Actividad presencial	3 hr. Pedagógicas	Trabajo autónomo	1,5 hr. Cronológicas
Fecha de inicio	14/08/2023			
Fecha de término	22/12/2023			

DESCRIPCIÓN DE LA ASIGNATURA

En esta asignatura, los estudiantes adquirirán habilidades fundamentales en el desarrollo de software utilizando el lenguaje de programación Python. A través de ejercicios prácticos y proyectos, los estudiantes aprenderán los conceptos básicos de la programación, estructuras de datos, lógica de programación y resolución de problemas. Explorarán temas como variables, bucles, condicionales, funciones y manipulación de datos. Al finalizar, los estudiantes serán capaces de crear programas simples, para futuros desarrollos en informática y programación.

RESULTADOS DE APRENDIZAJE

RA 1. Reconoce los fundamentos de programación requeridos para razonar, en forma lógica, problemas del ámbito de la ingeniería de base científica.

RA 2. Utiliza lenguajes de programación y herramientas computacionales para la resolución de problemas, utilizando el razonamiento lógico propio de las ingenierías de base científica.

UNIDADES DE APRENDIZAJE

Unidad 1: INTRODUCCION AL LENGUAJE DE PROGRAMACION PYTHON

- 1.1.- Entornos de programación
 - 1.1.1.- PYCHARM
 - 1.1.2.- VS CODE
 - 1.1.3.- Entorno del VS CODE

1.2.- Tipos de datos

- 1.2.1.- Tipo STRING (cadena de texto)
- 1.2.2.- Tipo INT (números enteros)
- 1.2.3.- Tipo FLOAT (números con coma)
- 1.2.4.- Tipo BOOL (valores booleanos)
- 1.2.5.- Tipo LISTAS (datos compuestos)

1.3.- Constantes y Variables

- 1.3.1.- Constantes
- 1.3.2.- Variables numéricas
- 1.3.3.- Variables de texto

1.4.- Operadores aritméticos

- 1.4.1.- Suma, resta, multiplicación y división
- 1.4.2.- Módulo, exponente, división baja

1.5.- Operadores de comparación

- 1.5.1.- Igual, distinto, menor que, menor igual que, mayor que, mayor o igual que

1.6.- Operadores lógicos

- 1.6.1.- AND, OR, NOT

Unidad 2: PROGRAMACION CON VARIABLES SIMPLES

2.1.- Entrada y salida de datos

- 2.1.1.- Sentencia INPUT
- 2.1.3.- Sentencia PRINT

2.2.- Sentencias Condicionales

- 2.2.1.- Sentencia IF ELSE
- 2.2.2.- Sentencia IF ELIF ELSE

2.3.- Sentencias de repetición

- 2.3.1.- Bucle FOR IN lista
- 2.3.2.- Bucle FOR IN ZIP lista
- 2.3.3.- Bucle FOR IN RANGE
- 2.3.4.- Bucle WHILE

Unidad 3: PROGRAMACION CON VARIABLES COMPUESTAS

3.1.- Estructuras de datos

- 3.1.1.- Listas
- 3.1.2.- Tuplas
- 3.1.3.- Métodos de Listas (len, append, insert, extend, pop, remove, clear, sort, reverse)

3.2.- Métodos de String (upper, lower, capitalize, find, index, isnumeric, isalpha, count, len, endswith, starwith, replace, split, substring).

3.3.- Funciones integradas

3.3.1.- max, min, round, bool, all, sum

3.4.- Construcción de Funciones

3.4.1.- Funciones simples sin argumentos

3.4.2.- Funciones con parámetros (argumentos)

3.4.3.- Funciones que retornen valores

3.4.4.- Funciones que retornen múltiples valores

3.4.5.- Llamada de funciones de otros módulos.

Unidad 4: MISCELANEOS

4.1.- Conjuntos

4.2.- Diccionarios

4.3.- Métodos de Diccionarios (keys, get, clear, pop, items)

METODOLOGÍA Y EVALUACIÓN

METODOLOGÍA:

Esta metodología se centra en una introducción gradual y práctica a los fundamentos de programación requeridos para razonar, en forma lógica, problemas del ámbito de la ingeniería de base científica, utilizando herramientas computacionales y el lenguaje de programación Python para la resolución de problemas. Comenzando con la familiarización con la sintaxis y los tipos de datos, los estudiantes avanzarán hacia la manipulación de variables simples y compuestas. A medida que adquieren habilidades, se les guiará en la creación de proyectos prácticos que incluyan análisis de datos básicos y la automatización de tareas, culminando en un trabajo final que les permitirá aplicar de manera integral lo aprendido. Las evaluaciones regulares y el repaso constante aseguran una comprensión sólida y un progreso constante a lo largo del curso.

EVALUACION:

RA1: 50%.

- Control 1: **80%**
- Quizes: **20%** (corresponde al promedio de los dos mejores quizes elegidos de entre los quizes 1, 2 y 3)

RA2: 50%

- Control 2: **60%**
- Quizes: **20%** (corresponde a la nota del mejor quiz elegido de entre los quizes 4 y 5)
- Trabajo final: **20%**

Nota Final = Nota RA1*0,5 + Nota RA2*0,5 , si Nota RA1 y Nota RA2 >= 4,0

(Si uno más RA's son reprobados, la Nota Final corresponderá a la nota más baja de los RA's)

BIBLIOGRAFÍA.

- **Python.org:** El sitio oficial de Python proporciona documentación detallada, tutoriales y recursos para aprender sobre el lenguaje y sus bibliotecas estándar. Puedes acceder a la documentación en <https://www.python.org/doc/> y encontrar tutoriales en <https://docs.python.org/3/tutorial/index.html>.
- **W3Schools - Python Tutorial:** W3Schools ofrece tutoriales en línea sobre varios temas, incluyendo Python. Puedes acceder a su tutorial en <https://www.w3schools.com/python/>.
- **Real Python:** Un sitio web con una variedad de artículos y tutoriales sobre Python, que cubren desde conceptos básicos hasta temas más avanzados. <https://realpython.com/>
- **GeeksforGeeks - Python Programming Language:** GeeksforGeeks ofrece tutoriales, ejemplos de código y desafíos de programación en Python. Puedes visitar su sección de Python en <https://www.geeksforgeeks.org/python-programming-language/>.