

PROGRAMA DE LABORATORIO

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS



INTRODUCCIÓN A LA PROGRAMACIÓN Y COMPUTACIÓN 2

CÓDIGO:	0771	PONDERACIÓN:	5 CREDITOS 6 CREDITOS CLAR
ESCUELA DE INGENIERÍA EN:	CIENCIAS Y SISTEMAS	ÁREA A LA QUE PERTENECE:	DESARROLLO DE SOFTWARE
PRE REQUISITO:	770- Introducción a la Programación y Computación I 107 - Matemática Intermedia 1 795 - Lógica Matemática 960 - Matemática de Cómputo 1	POST REQUISITO:	777 - Org. Lenguajes y Compiladores 1 964- Org. Computacional 772 - Estructura de Datos 601 - Inv. de operaciones 1
CATEGORÍA:	OBLIGATORIO	VIGENCIA:	SEGUNDO SEMESTRE 2025
HORAS POR SEMANA DEL CURSO:	3 horas y 20 minutos	HORAS POR SEMANA DEL LABORATORIO:	1 hora y 40 minutos
HORAS DE AUTOAPRENDIZAJE:	523	TOTAL DE HORAS DE APRENDIZAJE:	18 hora y 20 minutos
CATEDRÁTICO (A):	Ing. Dennis Stanley Barrios Gonzalez	AUXILIAR:	Luis Antonio Castillo Javier
EDIFICIO:	Meet	SECCIÓN:	D
SALÓN DEL CURSO:	Meet	SALON DEL LABORATORIO:	meet.google.com/akr-nu-mf-vrw
DÍAS QUE SE IMPARTE EL CURSO:	Sábado	DÍAS QUE SE IMPARTE EL LABORATORIO:	Martes
HORARIO DEL CURSO:	7:10 - 10:30	HORARIO DEL LABORATORIO:	17:20 - 19:00

Breve descripción del Laboratorio

El laboratorio del curso 'Introducción a la programación y computación 2' está diseñado para complementar la teoría impartida en clase, mediante la aplicación práctica de los conceptos fundamentales. Su propósito es fomentar el aprendizaje activo, el pensamiento computacional y el desarrollo de habilidades para la resolución de problemas usando métodos algorítmicos y herramientas actuales.

Índice

Competencias Vinculadas al Perfil del Egresado	4
Competencias Específicas	4
Competencias Generales	4
Competencias del Laboratorio	5
Competencia(s) Específica(s)	5
Competencia(s) General(es)	5
Diseño Didáctico por Competencias	6
Sesión de Diagnóstico	7
Evaluación de conocimientos previos	7
Presentación del tutor	7
Presentación de los estudiantes	7
Presentación del programa del curso	7
Evaluación de conocimientos del laboratorio actual	7
Sesión No. 1, Unidad No. 1 - Introducción a Python	8
Valor de la semana (Saber ser)	8
Conocimiento (Saber)	8
Habilidades (Saber Hacer)	8
Sesión No. 2, Unidad No. 2 - Introducción a Python, Manejo de versiones	9
Valor de la semana (Saber ser)	9
Conocimiento (Saber)	9
Habilidades (Saber Hacer)	9
Sesión No. 3, Unidad No. 3 - Procesamiento de datos XML	10
Valor de la semana (Saber ser)	10
Conocimiento (Saber)	10
Habilidades (Saber Hacer)	10
Sesión No. 4, Unidad No. 4 - Estructuras de Datos	11
Valor de la semana (Saber ser)	11
Conocimiento (Saber)	11
Habilidades (Saber Hacer)	11
Sesión No. 5, Unidad No. 4 - Estructuras de Datos	12
Valor de la semana (Saber ser)	12
Conocimiento (Saber)	12
Habilidades (Saber Hacer)	12
Sesión No. 6, Unidad No. 4 - Estructuras de Datos	14

Valor de la semana (Saber ser)	14
Conocimiento (Saber)	14
Habilidades (Saber Hacer)	14
Sesión No. 7 Unidad No. 5 - HTML5 y CSS	16
Valor de la semana (Saber ser)	16
Conocimiento (Saber)	16
Habilidades (Saber Hacer)	16
Sesión No. 8, Unidad No. 6 - Entorno para desarrollo Web en Python	17
Valor de la semana (Saber ser)	17
Conocimiento (Saber)	17
Habilidades (Saber Hacer)	17
Sesión No. 9, Unidad No. 6 - Entorno para desarrollo Web en Python,	18
Valor de la semana (Saber ser)	18
Conocimiento (Saber)	18
Habilidades (Saber Hacer)	18
Sesión No. 10, Unidad No. 7 - JavaScript Object Notation (JSON), Unidad No. 8 -Acceso a datos web	19
Valor de la semana (Saber ser)	19
Conocimiento (Saber)	19
Habilidades (Saber Hacer)	19
Sesión No. 11, Unidad No. 8 -Acceso a datos web	21
Valor de la semana (Saber ser)	21
Conocimiento (Saber)	21
Habilidades (Saber Hacer)	21
Tiempo de Auto-aprendizaje	23
Rúbrica de Evaluación	23
Resumen de Ponderaciones	23
Normativa Académica y Ética del Curso	24
Equipo Académico	25
Coordinador del Área	25
Sección A	25
Sección B	26
Sección C	27
Bibliografía	28
E-Grafía	28

Competencias Vinculadas al Perfil del Egresado

Competencias Específicas

No.	Competencia
1	Demuestra pensamiento crítico, actitud investigativa y rigor analítico en el planteamiento y la resolución de problemas complejos.
2	Aplica los conocimientos de su disciplina en la elaboración, fundamentación y defensa de argumentos para prevenir y resolver problemas complejos en su campo profesional, identificando y aplicando innovaciones.
3	Demuestra destreza y habilidad en la selección, uso y adaptación de herramientas metodológicas, tecnológicas, equipos especializados y en la lectura e interpretación de datos, pertinentes al contexto de su ejercicio profesional.
4	Toma decisiones profesionales con base en fundamentos teóricos, datos e información pertinente, válida y confiable.
5	Identifica oportunidades y riesgos para la innovación y adaptación de conocimientos y tecnologías para resolver problemas.

Competencias Generales

No.	Competencia
1	Aplica principios básicos de ingeniería, ciencias de computación y sistemas de información y comunicación, en la formulación y resolución adecuada de problemas complejos.
2	Maneja e Interpreta adecuadamente datos masivos, sean estos estructurados o no estructurados, facilitando su visualización e interpretación de forma eficaz en apoyo a la toma de decisiones.
3	Aplica conocimientos tecnológicos con ética profesional y respetando y cuidando los recursos naturales, humanos y financieros.
4	Aplica estándares de calidad, eficiencia y seguridad en la implementación adecuada de soluciones de software, hardware y TIC en general.
5	Actualiza permanente sus conocimientos relacionados con TIC en general, apoyándose en las estrategias de aprendizaje apropiadas.

Competencias del Laboratorio

Competencia(s) Específica(s)

No.	Competencia	Nivel de Aprendizaje
1	Crea estructuras de datos dinámicas (listas enlazadas, doblemente enlazadas y circulares) mediante POO en Python e integrado datos desde archivos XML para implementar sistemas de gestión de datos eficientes que manejen relaciones complejas	Crear
2	Desarrolla aplicaciones web con Flask o Django en Python para soportar arquitecturas backend y frontend bien definidas.	Desarrollar
3	Implementa comunicación cliente-servidor mediante protocolos HTTP/HTTPS para desarrollar APIs seguras y eficientes	Implementar
4	Crea páginas Web Básicas utilizando HTML5 y CSS para estructurar y dar estilo a componentes web.	Crear
5	Analiza actividades en archivos XML utilizando técnicas de lectura estructurada para identificar y extraer información relevante.	Aplicar
6	Aplica tuplas, listas, diccionarios nativos de Python en programas funcionales para almacenar y manipular conjunto de datos	Aplicar

Competencia(s) General(es)

No.	Competencia	Nivel de Aprendizaje
1	Aplica técnicas de control de versiones, como Git Flow, branching, merging y versionamiento semántico en GIT para asegurar la calidad del software mediante un historial de cambios estructurado.	Aplicar
2	Aplica estructuras de control y funciones mediante programación básica en Python. para desarrollar algoritmos que resuelvan problemas.	Aplicar

Diseño Didáctico por Competencias

Esta sección organiza las sesiones del laboratorio en función de las competencias que el estudiante debe desarrollar. Cada clase incluye valores (saber ser), contenidos teóricos (saber) y habilidades prácticas (saber hacer), permitiendo un aprendizaje integral y aplicado. Las actividades están alineadas con los objetivos del curso y el perfil del egresado.

Sesión de Diagnóstico

Evaluación de conocimientos previos

Se aplicará una actividad diagnóstica con el objetivo de identificar el nivel de conocimientos y habilidades que los estudiantes poseen al inicio del curso. No influye en la nota final, pero es obligatoria para todos los estudiantes.

Tipo de Actividad	Descripción
Cuestionario interactivo	Se utilizará kahoot para evaluar el conocimiento previo de los estudiantes.

Presentación del tutor

El tutor se presenta formalmente al grupo, compartiendo su formación académica, experiencia profesional y educativa, así como sus expectativas sobre el curso. También se abordan aspectos como normas de convivencia, canales de comunicación, disponibilidad para consultas y métodos de acompañamiento.

Presentación de los estudiantes

Se escogen un grupo de estudiantes al azar. En su presentación, se les pedirá que compartan información básica como su nombre, intereses personales o profesionales, experiencias previas relacionadas con el curso y sus expectativas. Esta actividad busca promover la interacción, el reconocimiento entre pares y la construcción de un entorno participativo y respetuoso.

Presentación del programa del curso

Se presenta el contenido del programa del curso, se aclaran dudas y se fomenta el compromiso del estudiante con su aprendizaje.

Evaluación de conocimientos del laboratorio actual

Se realiza una evaluación o práctica que permite conocer el grado de familiaridad de los estudiantes con las herramientas, entornos o competencias técnicas necesarias para el laboratorio actual.

Tipo de Actividad	Descripción
Ejercicios prácticos	Ejercicios cortos que engloben los temas vistos en el laboratorio usando el lenguaje de programación del curso

Sesión No. 1, Unidad No. 1 - Introducción a Python

Valor de la semana (Saber ser)

Nombre:
Respeto: "Todos deberían ser respetados como individuos, pero ninguno idealizado". Albert Einstein.

Conocimiento (Saber)

Competencia(s)	
Aplica estructuras de control y funciones mediante programación básica en Python. para desarrollar algoritmos que resuelvan problemas.	
Tema	Subtema
Introducción y Presentación	Instalación de Python 3.X.X
Introducción y Presentación	Instalación del entorno de desarrollo
Introducción y Presentación	Variables, Expresiones y sentencias
Introducción y Presentación	Ejecuciones condicionales
Introducción y Presentación	Funciones
Introducción y Presentación	Iteración
Introducción y Presentación	Strings

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Aplica estructuras de control y funciones mediante programación básica en Python. para desarrollar algoritmos que resuelvan problemas.	Ejercicio	0

Sesión No. 2, Unidad No. 2 - Introducción a Python, Manejo de versiones

Valor de la semana (Saber ser)

Nombre:
Responsabilidad: "No se puede escapar de la responsabilidad del mañana evadiéndola hoy". Abraham Lincoln

Conocimiento (Saber)

Competencia(s)	
Aplica estructuras de control y funciones mediante programación básica en Python. para desarrollar algoritmos que resuelvan problemas.	
Tema	Subtema
Programación Orientada a Objetos en Python	Introducción al paradigma de POO
Programación Orientada a Objetos en Python	Clases y Objetos(Métodos y Atributos)
Programación Orientada a Objetos en Python	Pilares de la POO (Herencia, Abstracción, Polimorfismo, Encapsulamiento)
Git Workflow	Introducción a GIT y Control de versiones
Git Workflow	Fundamentos de control de versiones con Git y su Configuración Inicial para el versionamiento de proyectos
Git Workflow	Git Flow: Implementación de Troncales y Ramas
Git Workflow	Branching, Merging y Utilización de Etiquetas (Tags) en Git

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Aplica estructuras de control y funciones mediante programación básica en Python. para desarrollar algoritmos que resuelvan problemas.	Ejercicio	0

Sesión No. 3, Unidad No. 3 - Procesamiento de datos XML

Valor de la semana (Saber ser)

Nombre:
"No podemos resolver problemas usando el mismo tipo de pensamientos que usamos cuando los creamos" - Albert Einstein

Conocimiento (Saber)

Competencia(s)	
Analiza actividades en archivos XML utilizando técnicas de lectura estructurada para identificar y extraer información relevante.	
Tema	Subtema
Lectura de actividades y XML	Introducción a bases de XML
Lectura de actividades y XML	Procesamiento de datos con librería DOM
Lectura de actividades y XML	Procesamiento de datos con librería ElementTree
Lectura de actividades y XML	Introducción Xpath
Lectura de actividades y XML	Procesamiento de datos con librería Xpath
Lectura de actividades y XML	Lectura XML
Lectura de actividades y XML	Escritura XML

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Analiza actividades en archivos XML utilizando técnicas de lectura estructurada para identificar y extraer información relevante.	Actividad	0

Sesión No. 4, Unidad No. 4 - Estructuras de Datos

Valor de la semana (Saber ser)

Nombre:
"En la vida no hay cosas que temer, solo cosas que comprender" - Marie Curie

Conocimiento (Saber)

Competencia(s)	
Aplica estructuras de control y funciones mediante programación básica en Python. para desarrollar algoritmos que resuelvan problemas.	
Tema	Subtema
Tuplas, Listas y Diccionarios Nativos de Python	Introducción estructuras de datos
Tuplas, Listas y Diccionarios Nativos de Python	Listas nativas de python
Tuplas, Listas y Diccionarios Nativos de Python	Ficheros
Tuplas, Listas y Diccionarios Nativos de Python	Diccionarios
Tuplas, Listas y Diccionarios Nativos de Python	Tuplas
Tuplas, Listas y Diccionarios Nativos de Python	Expresiones regulares
Tuplas, Listas y Diccionarios Nativos de Python	Introducción a TDA's

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Aplica estructuras de control y funciones mediante programación básica en Python. para desarrollar algoritmos que resuelvan problemas.	Actividad	0

Sesión No. 5, Unidad No. 4 - Estructuras de Datos

Valor de la semana (Saber ser)

Nombre:
Responsabilidad, Orden, Paciencia, Precisión, Pensamiento lógico

Conocimiento (Saber)

Competencia	
Crea estructuras de datos dinámicas (listas enlazadas, doblemente enlazadas y circulares) mediante POO en Python e integrado datos desde archivos XML para implementar sistemas de gestión de datos eficientes que manejen relaciones complejas	
Aplica técnicas de control de versiones, como Git Flow, branching, merging y versionamiento semántico en GIT para asegurar la calidad del software mediante un historial de cambios estructurado.	
Aplica estructuras de control y funciones mediante programación básica en Python. para desarrollar algoritmos que resuelvan problemas.	
Tema	Subtema
TDA	Concepto Lista enlazada simple
TDA	Operaciones en lista enlazada simple
TDA	Concepto Lista enlazada doble
TDA	Operaciones en lista enlazada doble
TDA	Concepto de lista circular
TDA	Operaciones en listas circulares
TDA	Diferencia entre lista circular y no circular

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Crea estructuras de datos dinámicas (listas enlazadas, doblemente enlazadas y circulares) mediante POO en Python e integrado datos desde archivos XML para implementar sistemas de gestión de datos eficientes que manejen relaciones complejas	Ejercicio	0
Crea estructuras de datos dinámicas (listas enlazadas, doblemente enlazadas y circulares) mediante POO en Python e integrado datos desde archivos XML para implementar sistemas de gestión de datos eficientes que manejen relaciones complejas	Cuestionario	0

Sesión No. 6, Unidad No. 4 - Estructuras de Datos

Valor de la semana (Saber ser)

Nombre:
Responsabilidad, Orden, Paciencia, Precisión, Pensamiento lógico

Conocimiento (Saber)

Competencia	
Crea estructuras de datos dinámicas (listas enlazadas, doblemente enlazadas y circulares) mediante POO en Python e integrado datos desde archivos XML para implementar sistemas de gestión de datos eficientes que manejen relaciones complejas	
Aplica técnicas de control de versiones, como Git Flow, branching, merging y versionamiento semántico en GIT para asegurar la calidad del software mediante un historial de cambios estructurado.	
Aplica estructuras de control y funciones mediante programación básica en Python. para desarrollar algoritmos que resuelvan problemas.	
Tema	Subtema
TDA y graphviz	Concepto de lista doblemente circular
TDA y graphviz	Operaciones básicas en lista doblemente circular
TDA y graphviz	Ventajas y comparación frente a otras listas
TDA y graphviz	Qué es Graphviz
TDA y graphviz	Cómo se usa Graphviz para TDA's
TDA y graphviz	Graficar TDA's con graphviz

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Crea estructuras de datos dinámicas (listas enlazadas, doblemente enlazadas y circulares) mediante POO en Python e integrado datos desde archivos XML para implementar sistemas de gestión de datos eficientes que manejen relaciones complejas	Ejercicio	0
Crea estructuras de datos dinámicas (listas enlazadas, doblemente enlazadas y circulares) mediante POO en Python e integrado datos desde archivos XML para implementar sistemas de gestión de datos eficientes que manejen relaciones complejas	Actividad	0

Sesión No. 7 Unidad No. 5 - HTML5 y CSS

Valor de la semana (Saber ser)

Nombre:
Honestidad: "La honestidad es el primer capítulo en el libro de la sabiduría." — Thomas Jefferson

Conocimiento (Saber)

Competencia	
Crea páginas Web Básicas utilizando HTML5 y CSS para estructurar y dar estilo a componentes web.	
Tema	Subtema
Introducción a HTML, CSS, JS	Introducción a HTML5
Introducción a HTML, CSS, JS	Estructura básica de HTML5
Introducción a HTML, CSS, JS	Componentes básicos de HTML5
Introducción a HTML, CSS, JS	Introducción a CSS
Introducción a HTML, CSS, JS	Estructura básica de CSS
Introducción a HTML, CSS, JS	Componentes básicos de CSS
Introducción a HTML, CSS, JS	Ejemplos prácticos

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Crea páginas Web Básicas utilizando HTML5 y CSS para estructurar y dar estilo a componentes web.	Actividad	0

Sesión No. 8, Unidad No. 6 - Entorno para desarrollo Web en Python

Valor de la semana (Saber ser)

Nombre:
Perseverancia: "La energía y la persistencia conquistan todas las cosas." — Benjamin Franklin

Conocimiento (Saber)

Competencia	
Desarrolla aplicaciones web con Flask o Django en Python para soportar arquitecturas backend y frontend bien definidas.	
Tema	Subtema
Introducción a servidores web y Flask	Entorno para desarrollo Web en Python
Introducción a servidores web y Flask	Flask
Introducción a servidores web y Flask	Introducción a Flask y Configuración del Entorno
Introducción a servidores web y Flask	Rutas y Vistas
Introducción a servidores web y Flask	Plantillas con Jinja
Introducción a servidores web y Flask	Manejo de Formularios y Datos de Usuario
Introducción a servidores web y Flask	Ejemplos prácticos

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Desarrolla aplicaciones web con Flask o Django en Python para soportar arquitecturas backend y frontend bien definidas.	Actividad	0

Sesión No. 9, Unidad No. 6 - Entorno para desarrollo Web en Python,

Valor de la semana (Saber ser)

Nombre:
Humildad: La humildad es saber que siempre hay algo nuevo que aprender

Conocimiento (Saber)

Competencia	
Desarrolla aplicaciones web con Flask o Django en Python para soportar arquitecturas backend y frontend bien definidas.	
Tema	Subtema
Instalación y Configuración en Django	Instalación y configuración
Instalación y Configuración en Django	Apps en Django
Instalación y Configuración en Django	Modelos en Django
Instalación y Configuración en Django	Creando un módulo
Creación de Vistas y Uso de Plantillas en Django	Creando Vistas
Creación de Vistas y Uso de Plantillas en Django	Panel de administración de Django
Creación de Vistas y Uso de Plantillas en Django	Templates y archivos estáticos en Django

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Desarrolla aplicaciones web con Flask o Django en Python para soportar arquitecturas backend y frontend bien definidas.	Ejercicio	0

Sesión No. 10, Unidad No. 7 - JavaScript Object Notation (JSON), Unidad No. 8 - Acceso a datos web

Valor de la semana (Saber ser)

Nombre:
Empatía: Ser empático es ver el mundo a través de los ojos del otro y no ver nuestro mundo reflejado en sus ojos

Conocimiento (Saber)

Competencia	
Crea páginas Web Básicas utilizando HTML5 y CSS para estructurar y dar estilo a componentes web.	
Analiza actividades en archivos XML utilizando técnicas de lectura estructurada para identificar y extraer información relevante.	
Tema	Subtema
Estructura y Manejo de Datos JSON	Sintaxis de JSON, JSON vs XML
Estructura y Manejo de Datos JSON	Tipos de datos JSON
Estructura y Manejo de Datos JSON	JSON Parse, Stringify, Objects, Arrays
Acceso y Manejo de Datos Web mediante HTTP en Python	Internet como origen de datos
Acceso y Manejo de Datos Web mediante HTTP en Python	Protocolo HTTP y Códigos de error
Acceso y Manejo de Datos Web mediante HTTP en Python	Peticiones HTTP en Python y librería en Python para realizar peticiones HTTP
Acceso y Manejo de Datos Web mediante HTTP en Python	Lectura de archivos binarios mediante peticiones HTTP y Análisis de XML en Peticiones

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Crea páginas Web Básicas utilizando HTML5 y CSS para estructurar y dar estilo a componentes web.	Ejercicio	0
Crea páginas Web Básicas utilizando HTML5 y CSS para estructurar y dar estilo a componentes web.	Actividad	0

Analiza actividades en archivos XML utilizando técnicas de lectura estructurada para identificar y extraer información relevante.	Práctica	0
---	----------	---

Sesión No. 11, Unidad No. 8 -Acceso a datos web

Valor de la semana (Saber ser)

Nombre:
Responsabilidad y compromiso con el aprendizaje activo, Colaboración en el trabajo en equipo, Perseverancia en la resolución de problemas técnicos.

Conocimiento (Saber)

Competencia	
Crea páginas Web Básicas utilizando HTML5 y CSS para estructurar y dar estilo a componentes web.	
Analiza respuestas de peticiones web mediante el uso de librerías HTTP y análisis JSON/XML para obtener datos seguros desde APIs externas.	
Tema	Subtema
Interacción y Seguridad en APIs y Datos Web	Desplazamiento a través de los nodos
Interacción y Seguridad en APIs y Datos Web	Análisis JSON en Peticiones
Interacción y Seguridad en APIs y Datos Web	Interfaces de programación de aplicaciones
Interacción y Seguridad en APIs y Datos Web	API
Interacción y Seguridad en APIs y Datos Web	Seguridad y uso de API's
Interacción y Seguridad en APIs y Datos Web	Autenticación y autorización en API's
Interacción y Seguridad en APIs y Datos Web	Manejo de respuestas asíncronas y promesas

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Crea páginas Web Básicas utilizando HTML5 y CSS para estructurar y dar estilo a componentes web.	Práctica	0
Analiza respuestas de peticiones web mediante el uso de librerías HTTP y análisis JSON/XML para obtener datos seguros desde APIs externas.	Ejercicio	0
Analiza respuestas de peticiones web mediante el uso de librerías HTTP y análisis JSON/XML para obtener datos seguros desde APIs externas.	Actividad	0

Tiempo de Auto-aprendizaje

Tipo	Horas de Auto-aprendizaje
Proyectos	300
Prácticas	223
Tareas	0
Total	523

Rúbrica de Evaluación

Cada una de las actividades del laboratorio (proyectos, prácticas, tareas y otras) cuenta con una rúbrica de evaluación específica, la cual está detallada en el documento que se entrega al estudiante al momento de asignar la actividad. Estas rúbricas describen los criterios de evaluación, niveles de desempeño esperados y la ponderación correspondiente de cada aspecto evaluado.

Es **responsabilidad del estudiante** leer detenidamente la rúbrica asignada antes de iniciar el desarrollo de la actividad. Comprender los criterios de evaluación no solo permite orientar adecuadamente el trabajo, sino también mejorar el desempeño académico y fomentar la autorregulación del aprendizaje.

En caso de no recibir la rúbrica al momento de la asignación, el estudiante **debe solicitarla directamente al tutor académico**, ya que constituye una herramienta esencial para el cumplimiento de los objetivos de aprendizaje y la evaluación transparente.

Resumen de Ponderaciones

Tipo	Valor
Actividades en Clase	0
Proyectos	90
Prácticas	0
Tareas	0
Examen Final	10
Total	100

Normativa Académica y Ética del Curso

En concordancia con el perfil del estudiante de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, se espera un alto nivel de compromiso con la excelencia académica y la ética profesional. Por ello, que se establece los siguientes lineamientos de carácter obligatorio que regulan el comportamiento académico del estudiante:

Plagio y copias

- Todo proyecto será sometido a verificación para confirmar su autoría y originalidad, con la finalidad de evitar cualquier plagio, copia o que la actividad no haya sido realizada por el estudiante.
- Cualquier evidencia de lo antes descrito en las distintas actividades será sancionada con una calificación de 0 (cero) y el caso será reportado al Docente quien a su vez informará a la Escuela de Ciencias y Sistemas para su seguimiento institucional.

Prórrogas y reposiciones

- No se otorgarán prórrogas para entregas de actividades.
- No se permitirá la reposición de proyectos bajo ninguna circunstancia.

Requisitos para evaluación final del curso

- Es obligatorio aprobar el laboratorio para tener derecho a la evaluación final del curso.
- La calificación de prácticas, proyectos y otras actividades que se indique será asignada de forma presencial, en la fecha y hora establecidas por el tutor académico.

Asistencia

- Para obtener la nota del laboratorio, se requiere un mínimo del 80% de asistencia a las sesiones de laboratorio.
- En caso de inasistencia, sólo se aceptarán justificaciones válidas respaldadas por constancia oficial.

Entregas

- No se aceptarán entregas tardías de tareas, prácticas, exámenes cortos, exámenes finales o proyectos sin justificación.

Medio oficial de entrega

- La plataforma UEDI de la Facultad será el único medio oficial para la entrega de actividades del curso.

Equipo Académico

Coordinador del Área

Nombre: Marlon Francisco Orellana Lopez	Correo electrónico: marlonorellana2005@gmail.com
---	--

Sección A

Docente

Nombre del Docente Marlon Antonio Pérez Türk	Correo electrónico 2346363510101@ingenieria.usac.edu.gt
--	---

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Día				X	X	
Horario				7:10 - 8:50	7:10 - 8:50	
Lugar				Meet	Meet	

Tutor(es)

Nombre del Tutor	Angely Naomi Marroquín Tapaz	
Correo electrónico institucional	3603870200101@ingenieria.usac.edu.gt	

Tipo		Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Clase	Día		X				
	Horario		17:20 - 19:00				
	Lugar		meet.google.com/cyp-trif-fvf				
Atención al Estudiante	Día						
	Horario						
	Lugar						

Sección B

Docente

Nombre del Docente Claudia Liceth Rojas Morales	Correo electrónico claudiarojasmorales@gmail.com
---	--

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Día				x	x	
Horario				7:10-8:50	7:10-8:50	
Lugar				MEET	MEET	

Tutor(es)

Nombre del Tutor	Diego Andres Huite Alvarez	
Correo electrónico institucional	3023135080101@ingenieria.usac.edu.gt	

Tipo		Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Clase	Día		x				
	Horario		17:20-19:00				
	Lugar		https://meet.google.com/i-oy-nqaj-vjd				
Atención al Estudiante	Día						
	Horario						
	Lugar						

Sección C

Docente

Nombre del Docente José Manuel Ruíz Juárez	Correo electrónico 1911469000101@ingenieria.usac.edu.gt
--	---

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Día				X	X	
Horario				7:10 - 8:50	7:10 - 8:50	
Lugar				Meet	Meet	

Tutor(es)

Nombre del Tutor	Hesban Amilcar Argueta Aguilar	
Correo electrónico institucional	1649841270115@ingenieria.usac.edu.gt	

Tipo		Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Clase	Día		X				
	Horario		17:20-19:00				
	Lugar		meet.google.com/wig-fnh-m-wuy				
Atención al Estudiante	Día						
	Horario						
	Lugar						

Sección D

Docente

Nombre del Docente Dennis Stanley Barrios Gonzalez	Correo electrónico stanley.barrios@gmail.com
---	---

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Día						X
Horario						7:10-10:30
Lugar						Meet

Tutor(es)

Nombre del Tutor	Luis Antonio Castillo Javier	
Correo electrónico institucional	3020186890101@ingenieria.usac.edu.gt	

Tipo		Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Clase	Día		X				
	Horario		17:20-19:00				
	Lugar		meet.google.com/akr-numf-vrw				
Atención al Estudiante	Día						
	Horario						
	Lugar						

Sección N

Docente

Nombre del Docente Edwin Estuardo Zapeta Gomez	Correo electrónico estuardo.zapeta@gmail.com
--	--

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Día				X	X	
Horario				17:20 - 19:00	17:20 - 19:00	
Lugar				Meet	Meet	

Tutor(es)

Nombre del Tutor	Pedro Luis Pu Tavico	
Correo electrónico institucional	3004359830101@ingenieria.usac.edu.gt	

Tipo		Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Clase	Día		X				
	Horario		17:20 - 19:00				
	Lugar		meet.google.com/vuc-vcww-fuz				
Atención al Estudiante	Día						
	Horario						
	Lugar						

Sección P

Docente

Nombre del Docente Fernando José Paz González	Correo electrónico fernandopaz1995@gmail.com
---	--

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Día	X		X			
Horario	17:20 - 19:00		17:20 - 19:00			
Lugar	Meet		Meet			

Tutor(es)

Nombre del Tutor	Angel Miguel García Urizar	
Correo electrónico institucional	3046108010115@ingenieria.usac.edu.gt	

Tipo		Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Clase	Día		X				
	Horario		17:20 - 19:00				
	Lugar		meet.google.com/iaw-wtyr-fra				
Atención al Estudiante	Día						
	Horario						
	Lugar						

Evaluación de rendimiento académico

Procedimiento de evaluación		Ponderación	Envío enunciado	Entrega
Proyecto No. I	Entregable 1 (30%)	9 pts.	04/08/2025	29/08/2025
Proyecto No. II	Entregable 2 (30%)	9 pts.	01/09/2025	26/09/2025
Proyecto No. III	Entregable 3 (30%)	9 pts.	27/09/2025	24/10/2025
Evaluación final	(10%)	3 pts.		
Total del laboratorio (100%)		30 pts.		
Nota de promoción		18.3 pts.		

Bibliografía

- Python para informáticos Versión 2.7.2 Charles Severance
- Fundamentos de programación, Algoritmos, Estructuras de Datos y Objetos, Luis Joyanes Aguilar, Cuarta Edición, McGraw-Hill
- The Scrum Guide™ Ken Schwaber and Jeff Sutherland. 2017
- Craig Larman; Introducción al análisis y diseño orientado a objetos. Prentice Hall
- Software Quality Assurance, From Theory to Implementation, Daniel Galin 2004.
- State of Software Development, CodingSans 2019
- El control de versiones, Guillem Borrell, 2006.
- Software Design, David Budgen, 2ª. Edición, Pearson Addison Wesley.
- OWASP Application Security Verification Standard 4.0, 2019

E-Grafía