

PROGRAMA DE LABORATORIO

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



FIUSAC
FACULTAD DE INGENIERÍA
UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

**LABORATORIO INTRODUCCIÓN A LA PROGRAMACIÓN Y COMPUTACIÓN 1 -
SECCIÓN B**

CÓDIGO:	770	PUNTEO NETO LABORATORIO:	20
ESCUELA DE INGENIERÍA EN:	CIENCIAS Y SISTEMAS	ÁREA A LA QUE PERTENECE:	DESARROLLO DE SOFTWARE
PRE REQUISITO:	103 - ÁREA MATEMÁTICA BÁSICA 2 147 - FÍSICA BÁSICA 960 - MATEMÁTICA PARA COMPUTACIÓN 1 768 - ALGORITMOS	POST REQUISITO:	796 - LENGUAJES FORMALES Y DE PROGRAMACIÓN 771 - INTRODUCCIÓN A LA PROGRAMACIÓN Y COMPUTACIÓN 2 2025 - PRÁCTICAS INICIALES
CATEGORÍA:	OBLIGATORIO	VIGENCIA:	PRIMER SEMESTRE 2026

Descripción del Laboratorio

El curso busca ser el acercamiento inicial del estudiante de la carrera de sistemas, al mundo de Desarrollo de Software mediante el uso de métodos, técnicas y metodologías especializadas. Se fundamenta en el concepto de algoritmo para la resolución de problemas de programación utilizando computadoras, enfatizando el uso del paradigma de Programación Orientado a Objetos. Se acerca al estudiante al conocimiento de los principales algoritmos de búsquedas y ordenamientos. Asimismo, el estudiante conocerá el lenguaje Java como el lenguaje oficial de programación del curso.

Resumen de Ponderaciones y Tiempo de Auto-aprendizaje

TIPO	PONDERACIÓN	HORAS DE AUTO-APRENDIZAJE
Actividades en Clase	20	0
Proyectos	40	134
Prácticas	20	84
Tareas	10	0
Examen Final	10	0
TOTAL	100	218

Equipo Académico

Coordinador del Área

Nombre: MARLON FRANCISCO ORELLANA LOPEZ	Correo electrónico: marlonorellana2005@gmail.com
---	--

Docente

Nombre del Docente	Correo electrónico del Docente
WILLIAM ESTUARDO ESCOBAR ARGUETA	guatesistemas@gmail.com

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Día		X		X		
Horario		7:10 - 8:50		7:10 - 8:50		
Lugar		T3 - 309		T3 - 309		

Tutor(es)

Nombre del Tutor	Fernando José Vicente Velásquez	
Correo electrónico institucional	3021894750101@ingenieria.usac.edu.gt	

Tipo		Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Clase	Día						X
	Horario						10:30 am - 12:10 pm
	Lugar						T3 - 310
Atención al Estudiante	Día						X
	Horario						10:30 am - 12:10 pm
	Lugar						T3 - 310

Índice

Descripción del Laboratorio	1
Resumen de Ponderaciones y Tiempo de Auto-aprendizaje	1
Equipo Académico	2
Coordinador del Área	2
Docente	2
Tutor(es)	2
Competencias Vinculadas al Perfil del Egresado	4
Competencias Específicas	4
Competencias Generales	4
Competencias del Laboratorio	4
Competencia(s) Específica(s)	4
Competencia(s) General(es)	4
Diseño Didáctico	5
Sesión de Diagnóstico	5
Sesión No. 2: Unidad No. 1 - Lógica Algorítmica y Diagramas de Flujo	6
Sesión No. 3: Unidad No. 1 - Lógica en Pseudocódigo y Control de Versiones con Git	7
Sesión No. 4: Unidad No. 2 - Entorno de Desarrollo y Sintaxis Básica en Java	8
Sesión No. 5: Unidad No. 2 - Estructuras de Control, Manejo de Errores y Arreglos	9
Sesión No. 6: Unidad No. 3 - Recursividad, Métodos de Ordenamiento y Archivos Planos	10
Sesión No. 7: Unidad No. 4 - Abstracción y Pilares Fundamentales de la POO	11
Sesión No. 8: Unidad No. 5 - Introducción a Interfaces Gráficas (Swing) y Arquitectura MVC	12
Sesión No. 9: Unidad No. 4 - Serialización de Objetos y Programación Multihilo	13
Sesión No. 10: Unidad No. 5 - Gestión de Eventos y Componentes Avanzados de Interfaz	14
Sesión No. 11: Unidad No. 6 - Modelado de Clases UML y Fundamentos de Cloud Computing	15
Rúbrica de Evaluación	16
Normativa Académica y Ética del Curso	16
Bibliografía	17
E-Grafía	17

Competencias Vinculadas al Perfil del Egresado

Competencias Específicas

No.	Competencia
1	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.
2	Identifica oportunidades y riesgos para la innovación y adaptación de conocimientos y tecnologías para resolver problemas.
3	Demuestra pensamiento crítico, actitud investigativa y rigor analítico en el planteamiento y la resolución de problemas complejos.

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	Aplica estándares de calidad, eficiencia y seguridad en la implementación adecuada de soluciones de software, hardware y TIC en general.
3	Aplica conocimientos tecnológicos con ética profesional y respetando y cuidando los recursos naturales, humanos y financieros.

Competencias del Laboratorio

Competencia(s) Específica(s)

No.	Competencia	Nivel de Aprendizaje
1	Analiza Estructuras de datos complejas y algoritmos de ordenamiento Utilizando técnicas de análisis de complejidad temporal y Para optimizar el rendimiento y eficiencia de aplicaciones	Evaluar
2	Implementa Sistemas de software con persistencia de datos Utilizando bases de datos relacionales, archivos estructurados y APIs Conforme a especificaciones de integridad, consistencia y seguridad	Aplicar
3	Evaluá Arquitecturas de software y calidad de código Empleando métricas de software, patrones de diseño y técnicas de testing Para garantizar mantenibilidad, escalabilidad y robustez del sistema	Evaluar

4	Diseña Interfaces gráficas de usuario intuitivas y funcionales Aplicando frameworks de desarrollo GUI, principios UX/UI y manejo de eventos Considerando usabilidad, accesibilidad y experiencia del usuario final	Crear
---	--	-------

Competencia(s) General(es)

No.	Competencia	Nivel de Aprendizaje
1	Colabora En equipos multidisciplinarios de desarrollo de software Aplicando metodologías ágiles, herramientas de gestión y comunicación efectiva Adaptando el nivel de detalle según el contexto y audiencia objetivo	Evaluar
2	Comunica Soluciones técnicas complejas a audiencias diversas Utilizando documentación técnica, presentaciones y demostraciones interactivas Adaptando el nivel de detalle según el contexto y audiencia objetivo	Evaluar
3	Reflexiona Críticamente sobre el impacto ético y social de las soluciones tecnológicas Considerando principios de responsabilidad social, privacidad y sostenibilidad Para tomar decisiones técnicas éticamente informadas y responsables	Evaluar

Diseño Didáctico

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
Examen corto en Quizziz	Corto Quizziz sobre temas abarcados en cursos anteriores que puedan llegar a tener una cierta similitud con el contenido del curso actual.

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 comparten 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
Examen corto en Quizziz	Se hará una corta prueba en quizziz con temas que se abordarán a lo largo que se desarrolle el laboratorio.

Sesión No. 2: Unidad No. 1 - Lógica Algorítmica y Diagramas de Flujo.

Área Actitudinal (Saber ser)

Nombre del valor:
Puntualidad

Área de Conocimiento (Saber)

Competencia(s)	
Comunica soluciones técnicas y analiza estructuras de datos complejas para optimizar rendimiento.	
Tema	Subtema
Fundamentos de Programación	Algoritmos
	Diagramas de Flujo

Área de Habilidades (Saber Hacer)

Competencia	
Analiza Estructuras de datos complejas y algoritmos de ordenamiento.	
Tipo de Actividad	Ponderación
Diseño de algoritmos con diagramas de flujo y pseudocódigo.	1%

Sesión No. 3: Unidad No. 1 - Lógica en Pseudocódigo y Control de Versiones con Git.

Área Actitudinal (Saber ser)

Nombre del valor:
Responsabilidad

Área de Conocimiento (Saber)

Competencia(s)	
Comunica soluciones técnicas complejas a audiencias diversas utilizando documentación técnica.	
Tema	Subtema
Introducción al versionamiento y lógica básica	Pseudocódigo, Git, Herramientas de control de versiones y Repositorios remotos .

Área de Habilidades (Saber Hacer)

Competencia	
Comunica Soluciones técnicas complejas a audiencias diversas.	
Tipo de Actividad	Ponderación
Resolución de problemas lógicos mediante pseudocódigo en Pseint.	0.5%

Sesión No. 4: Unidad No. 2 - Entorno de Desarrollo y Sintaxis Básica en Java.

Área Actitudinal (Saber ser)

Nombre del valor:
Disciplina

Área de Conocimiento (Saber)

Competencia(s)	
Comunica soluciones técnicas complejas utilizando demostraciones interactivas.	
Tema	Subtema
Introducción a Java	Versiones y ambientes (JDK, JRE, JVM), Variables, Tipos primitivos, Casteos y Operadores.

Área de Habilidades (Saber Hacer)

Competencia	
Comunica Soluciones técnicas complejas a audiencias diversas.	
Tipo de Actividad	Ponderación
Creación de repositorio en GitHub y ejercicios de sintaxis básica.	1%

Sesión No. 5: Unidad No. 2 - Estructuras de Control, Manejo de Errores y Arreglos.

Área Actitudinal (Saber ser)

Nombre del valor:
Honestidad

Área de Conocimiento (Saber)

Competencia(s)	
Evalúa arquitecturas de software y calidad de código para garantizar robustez.	
Tema	Subtema
Introducción a Java - Estructuras de Control y Memoria	Ciclos, Try-catch-finally, Procedimientos y Funciones, Debugging, Arreglos y Matrices .

Área de Habilidades (Saber Hacer)

Competencia	
Comunica Soluciones técnicas complejas y evalúa arquitecturas de software.	
Tipo de Actividad	Ponderación
Creación de una calculadora de consola con menú de operaciones.	1%

Sesión No. 6: Unidad No. 3 - Recursividad, Métodos de Ordenamiento y Archivos Planos.

Área Actitudinal (Saber ser)

Nombre del valor:
Honestidad

Área de Conocimiento (Saber)

Competencia(s)	
Analiza estructuras de datos y algoritmos de ordenamiento mediante técnicas de complejidad temporal.	
Tema	Subtema
Recursividad y Archivos.	Métodos de ordenamiento, Pila de ejecución (Call Stack), Recursividad y Manejo de archivos planos.

Área de Habilidades (Saber Hacer)

Competencia	
Analiza Estructuras de datos complejas y algoritmos de ordenamiento.	
Tipo de Actividad	Ponderación
Desglose práctico del algoritmo de ordenamiento burbuja.	0% (Actividad de aprendizaje)

Sesión No. 7: Unidad No. 4 - Abstracción y Pilares Fundamentales de la POO.

Área Actitudinal (Saber ser)

Nombre del valor:
Honestidad

Área de Conocimiento (Saber)

Competencia(s)	
Evalúa arquitecturas de software y calidad de código empleando métricas y patrones	
Tema	Subtema
Introducción a la Programación Orientada a Objetos (POO).	Clases y objetos, Pilares de POO, Miembros de clase, Constructores y Modificadores de visibilidad

Área de Habilidades (Saber Hacer)

Competencia	
Evalúa Arquitecturas de software y calidad de código.	
Tipo de Actividad	Ponderación
Cuestionario sobre conceptos básicos y fragmentos de código POO.	0%

Sesión No. 8: Unidad No. 5 - Introducción a Interfaces Gráficas (Swing) y Arquitectura MVC.

Área Actitudinal (Saber ser)

Nombre del valor:
Disciplina

Área de Conocimiento (Saber)

Competencia(s)	
Diseña interfaces gráficas intuitivas y evalúa arquitecturas mediante patrones de diseño.	
Tema	Subtema
Interfaces Gráficas y Patrones de Diseño.	Elementos de Swing/AWT y Patrón Modelo-Vista-Controlador (MVC).

Área de Habilidades (Saber Hacer)

Competencia	
Diseña Interfaces gráficas de usuario intuitivas y funcionales.	
Tipo de Actividad	Ponderación
Diseño de ventanas Swing aplicando el patrón de diseño MVC.	0%

Sesión No. 9: Unidad No. 4 - Serialización de Objetos y Programación Multihilo.

Área Actitudinal (Saber ser)

Nombre del valor:
Responsabilidad

Área de Conocimiento (Saber)

Competencia(s)	
Evalúa arquitecturas de software para garantizar mantenibilidad y escalabilidad.	
Tema	Subtema
POO Avanzado e Hilos.	Clases abstractas e Interfaces, Miembros estáticos, Serialización de objetos y Manejo de Hilos .

Área de Habilidades (Saber Hacer)

Competencia	
Analiza Estructuras de datos complejas y algoritmos de ordenamiento.	
Tipo de Actividad	Ponderación
Creación de una aplicación en consola multihilo con ciclo de vida.	0%

Sesión No. 10: Unidad No. 5 - Gestión de Eventos y Componentes Avanzados de Interfaz.

Área Actitudinal (Saber ser)

Nombre del valor:
Responsabilidad

Área de Conocimiento (Saber)

Competencia(s)	
Diseña interfaces gráficas aplicando frameworks GUI y manejo de eventos.	
Tema	Subtema
Interfaces Gráficas de Usuario (GUI) Avanzadas.	Componentes de AWT y Swing, Layout Managers y Manejo de eventos.

Área de Habilidades (Saber Hacer)

Competencia	
Diseña interfaces gráficas intuitivas y funcionales.	
Tipo de Actividad	Ponderación
Creación de una calculadora gráfica avanzada usando elementos Swing.	0%

Sesión No. 11: Unidad No. 6 - Modelado de Clases UML y Fundamentos de Cloud Computing.

Área Actitudinal (Saber ser)

Nombre del valor:
Disciplina

Área de Conocimiento (Saber)

Competencia(s)	
Evalúa arquitecturas de software e implementa sistemas con persistencia de datos.	
Tema	Subtema
UML y Cloud Computing	Diagramas de clases, Cloud Computing (Tipos y Modelos: SaaS, PaaS, IaaS) y Proveedores

Área de Habilidades (Saber Hacer)

Competencia	
Analiza Estructuras de datos complejas y algoritmos de ordenamiento.	
Tipo de Actividad	Ponderación
Creación de modelo de datos de un sistema mediante diagramas de clases UML.	0%

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.

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.

Bibliografía

-

E-Grafía

-