

LABORATORIO INTRODUCCIÓN A LA PROGRAMACIÓN Y COMPUTACIÓN 1 - SECCIÓN C

CÓDIGO:	770	PUNTEO NETO LABORATORIO:	X
ESCUELA DE INGENIERÍA EN:	CIENCIAS Y SISTEMAS	ÁREA A LA QUE PERTENECE:	DESARROLLO DE SOFTWARE
PRE REQUISITO:	768 - INTRODUCCIÓN A ALGORITMOS 103 - ÁREA MATEMÁTICA BÁSICA 2 147 - FÍSICA BÁSICA 960 - MATEMÁTICA PARA COMPUTACIÓN 1	POST REQUISITO:	771 - INTRODUCCIÓN A LA PROGRAMACIÓN Y COMPUTACIÓN 2 964 - ORGANIZACIÓN COMPUTACIONAL 2025 - PRÁCTICAS INICIALES
CATEGORÍA:	OBLIGATORIO	VIGENCIA:	SEGUNDO SEMESTRE 2026

Descripción del Laboratorio

El laboratorio del curso "Introducción a la Programación y Computación 1" tiene como propósito consolidar los conocimientos de lógica algorítmica adquiridos en el curso prerequisite Introducción a los Algoritmos y Flujo de Datos, orientándolos hacia el desarrollo de aplicaciones de software mediante el uso de un lenguaje de programación y herramientas propias del entorno profesional.

Durante el laboratorio, el estudiante aplicará los principios del pensamiento computacional y el diseño algorítmico para implementar soluciones en el lenguaje Java, fortaleciendo competencias relacionadas con la programación estructurada y la Programación Orientada a Objetos (POO). Asimismo, adquirirá experiencia en el uso de herramientas de control de versiones con Git, el modelado básico de soluciones mediante UML, la construcción de interfaces gráficas utilizando Swing bajo la arquitectura Modelo-Vista-Controlador (MVC) y el desarrollo de aplicaciones que integren conceptos como manejo de archivos, serialización de objetos y programación concurrente.

Resumen de Ponderaciones y Tiempo de Auto-Aprendizaje

TIPO	CANTIDAD	PONDERACIÓN	HORAS DE AUTO-APRENDIZAJE
Actividades en Clase	2	10	5
Proyectos	2	55	134

Prácticas	2	20	84
Tareas	5	5	5
Examen Final	1	10	0
TOTAL	X	100	218

Desglose de Actividades

TIPO DE ACTIVIDAD	PONDERACIÓN	HORAS DE AUTO-APRENDIZAJE
Proyecto 1	25	64
Proyecto 2	30	70
Práctica 1	10	42
Práctica 2	10	42
Actividad en Clase (2)	10 (5)	5
Tareas (5)	5 (1)	5
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 MOISÉS EDUARDO VELÁSQUEZ OLIVA	Correo electrónico del Docente mevo284@gmail.com
--	---

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

Tutor

Nombre del Tutor	Elías Abraham Vasquez Soto	
Correo electrónico institucional	3034289450108@ingenieria.usac.edu.gt	

Tipo		Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Clase	Día						X
	Horario						10:30 am - 12:10 pm
	Lugar						Salón 210, T-3
Atención al Estudiante	Día						X
	Horario						10:30 am - 12:10 pm
	Lugar						Salón 210, T-3

Índice

Descripción del Laboratorio.....	1
Resumen de Ponderaciones y Tiempo de Auto-Aprendizaje.....	1
Desglose de Actividades.....	2
Equipo Académico.....	2
Coordinador del Área.....	2
Docente.....	2
Tutor.....	3
Competencias Vinculadas al Perfil del Egresado.....	5
Competencias Específicas.....	5
Competencias Generales.....	5
Competencias del Laboratorio.....	5
Competencia(s) Específica(s).....	5
Competencia(s) General(es).....	6
Diseño Didáctico.....	6
Sesión de Diagnóstico.....	6
Sesión No. 2, Unidad No. 1 - Fundamentos de Programación.....	7
Sesión No. 3, Unidad No. 1 - Fundamentos de Programación.....	8
Sesión No. 4, Unidad No. 2 - Java Básico.....	9
Sesión No. 5, Unidad No. 2 - Java Básico.....	10
Sesión No. 6, Unidad No. 3 - Java Avanzado.....	11
Sesión No. 7, Unidad No. 4 - Abstracción y Pilares Fundamentales de la POO.....	12
Sesión No. 8, Unidad No. 5 - Introducción a Interfaces Gráficas (Swing) y Arquitectura MVC.....	13
Sesión No. 9, Unidad No. 4 - POO.....	14
Sesión No. 10, Unidad No. 5 - Gestión de Eventos y Componentes Avanzados de Interfaz.....	15
Sesión No. 11, Unidad No. 6 - Modelado de Clases UML y Fundamentos de Cloud Computing.....	16
Rúbrica de Evaluación.....	17
Normativa Académica y Ética del Curso.....	17
Bibliografía.....	18
E-Grafía.....	18

Competencias Vinculadas al Perfil del Egresado

Competencias Específicas

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

Competencias Generales

No.	Competencia
1	Actualiza permanentemente sus conocimientos relacionados con TIC en general, apoyándose en las estrategias de aprendizaje apropiadas.
2	Construye soluciones integrales trabajando en forma colaborativa y propositiva en equipos interdisciplinarios, en forma presencial o utilizando plataformas virtuales.
3	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.

Competencias del Laboratorio

Competencia(s) Específica(s)

No.	Competencia	Nivel de Aprendizaje
1	Analiza Estructuras de datos y algoritmos utilizando técnicas de análisis de complejidad temporal, con el objetivo de optimizar el rendimiento y eficiencia de las aplicaciones	Evaluar
2	Implementa sistemas de software en Java utilizando archivos y APIs, aplicando principios de integridad y seguridad de los datos para desarrollar soluciones confiables y robustas.	Aplicar
3	Evalúa Arquitecturas de software mediante la aplicación de patrones de diseño y técnicas de testing para desarrollar soluciones mantenibles, robustas y de calidad.	Evaluar

4	Diseña interfaces gráficas de usuario mediante el uso de frameworks GUI y la aplicación de principios de UX/UI para desarrollar aplicaciones usables, accesibles y centradas en el usuario.	Crear
5	Colabora en proyectos de desarrollo de software utilizando sistemas de control de versiones (Git) y metodologías ágiles, promoviendo la comunicación efectiva, la coordinación y el trabajo en equipo.	Aplicar

Competencia(s) General(es)

No.	Competencia	Nivel de Aprendizaje
1	Colabora en equipos multidisciplinarios de desarrollo de software mediante la aplicación de metodologías ágiles, herramientas de gestión y estrategias de comunicación efectiva para contribuir al desarrollo organizado y eficiente de soluciones informáticas.	Evaluar
2	Comunica soluciones técnicas a audiencias diversas mediante documentación técnica, presentaciones y demostraciones, adaptando el nivel de detalle según el contexto y la audiencia objetivo para facilitar la comprensión y el intercambio de conocimiento.	Evaluar
3	Reflexiona críticamente sobre el impacto ético y social de las soluciones tecnológicas considerando principios de responsabilidad profesional, privacidad, seguridad y sostenibilidad para tomar decisiones técnicas 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 el curso anterior de algoritmos que son fundamentales para entender 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 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
Examen corto en Quizziz	Se hará una corta prueba en Quizziz con temas que se abordarán a lo largo del desarrollo del laboratorio.

Sesión No. 2, Unidad No. 1 - Fundamentos de Programación

Área Actitudinal (Saber ser)

Nombre del valor:
Puntualidad

Área de Conocimiento (Saber)

Competencia(s)
Comprende los fundamentos del diseño algorítmico y la representación de soluciones mediante diagramas de flujo como etapa previa a la implementación de sistemas de software.

Tema	Subtema
Lógica algorítmica y Diagramas de Flujo	Algoritmos
	Diagramas de flujo

Área de Habilidades (Saber Hacer)

Competencia	
Diseña y comunica soluciones a problemas computacionales mediante algoritmos y diagramas de flujo, aplicando buenas prácticas de organización lógica y documentación técnica.	
Tipo de Actividad	Ponderación
Diseño de algoritmos con diagramas de flujo y pseudocódigo.	0

Sesión No. 3, Unidad No. 1 - Fundamentos de Programación

Área Actitudinal (Saber ser)

Nombre del valor:
Responsabilidad

Área de Conocimiento (Saber)

Competencia(s)	
Comprende los fundamentos del control de versiones, el funcionamiento de Git, los repositorios remotos y su importancia para la gestión de proyectos y el desarrollo colaborativo de software.	
Tema	Subtema
Versionamiento	Git, Herramientas de control de versiones y Repositorios remotos

Área de Habilidades (Saber Hacer)

Competencia	
Utiliza Git y repositorios remotos para gestionar versiones de un proyecto de software, aplicando operaciones básicas de control de versiones y buenas prácticas de trabajo colaborativo.	
Tipo de Actividad	Ponderación
Configuración del primer repositorio en GitHub y comandos básicos (push/pull).	0

Sesión No. 4, Unidad No. 2 - Java Básico

Área Actitudinal (Saber ser)

Nombre del valor:
Disciplina

Área de Conocimiento (Saber)

Competencia(s)	
Comprende la arquitectura y el entorno de ejecución de Java, diferenciando los componentes JDK, JRE y JVM, así como los conceptos fundamentales de variables, tipos de datos primitivos, conversiones de tipo y operadores para la implementación de programas.	
Tema	Subtema
Introducción a Java - Sintaxis Inicial	Versiones y ambientes (JDK, JRE, JVM), Variables, Tipos primitivos, Casteos y Operadores.

Área de Habilidades (Saber Hacer)

Competencia	
Implementa programas básicos en Java utilizando variables, tipos de datos primitivos, conversiones de tipo y operadores, aplicando buenas prácticas de codificación y aprovechando adecuadamente el entorno de desarrollo.	
Tipo de Actividad	Ponderación
CORTO 1 - Programa de consola: Conversor de bases numéricas y tipos primitivos.	5

Sesión No. 5, Unidad No. 2 - Java Básico

Área Actitudinal (Saber ser)

Nombre del valor:
Honestidad

Área de Conocimiento (Saber)

Competencia(s)	
Comprende el funcionamiento de las estructuras de control, el manejo de excepciones, la modularización mediante procedimientos y funciones, así como el uso de arreglos y matrices para el desarrollo de programas en Java.	
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	
Implementa soluciones en Java utilizando estructuras de control, procedimientos, funciones, manejo básico de excepciones, arreglos y matrices, empleando herramientas de depuración para verificar y corregir el funcionamiento de los programas.	
Tipo de Actividad	Ponderación
Desarrollo de calculadora científica con manejo de excepciones (Try-Catch).	0

Sesión No. 6, Unidad No. 3 - Java Avanzado

Área Actitudinal (Saber ser)

Nombre del valor:
Perseverancia

Área de Conocimiento (Saber)

Competencia(s)	
Comprende los principios de la recursividad, el funcionamiento de la pila de ejecución (Call Stack), los métodos básicos de ordenamiento y el manejo de archivos planos como mecanismos para el procesamiento y almacenamiento de información en aplicaciones Java.	
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	
Implementa soluciones en Java utilizando algoritmos recursivos, métodos básicos de ordenamiento y operaciones de lectura y escritura de archivos planos para resolver problemas de procesamiento y persistencia de datos.	
Tipo de Actividad	Ponderación
Implementación de algoritmos de ordenamiento (Burbuja) en arreglos.	0

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

Área Actitudinal (Saber ser)

Nombre del valor:
Integridad

Área de Conocimiento (Saber)

Competencia(s)	
Comprende los fundamentos de la Programación Orientada a Objetos, identificando el propósito de las clases, objetos, constructores, miembros de clase, modificadores de visibilidad y los pilares fundamentales de la orientación a objetos para el desarrollo de aplicaciones en Java.	
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	
Implementa soluciones en Java aplicando los principios fundamentales de la Programación Orientada a Objetos mediante la definición de clases, creación de objetos, uso de constructores y encapsulamiento de datos, siguiendo buenas prácticas de diseño de software.	
Tipo de Actividad	Ponderación
Modelado de un sistema de ventas utilizando Clases, Atributos y Métodos.	0

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

Área Actitudinal (Saber ser)

Nombre del valor:
Innovación

Área de Conocimiento (Saber)

Competencia(s)	
Comprende los fundamentos del desarrollo de interfaces gráficas de usuario utilizando los componentes de Swing/AWT y el patrón arquitectónico Modelo-Vista-Controlador (MVC), identificando su contribución a la organización, usabilidad y mantenibilidad de las aplicaciones.	
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 mediante el uso de componentes de Swing/AWT y la aplicación del patrón Modelo-Vista-Controlador (MVC), favoreciendo la organización del código y una adecuada interacción entre el usuario y la aplicación.	
Tipo de Actividad	Ponderación
Aplicación del patrón MVC para separar lógica de negocio de la GUI Swing.	0

Sesión No. 9, Unidad No. 4 - POO

Área Actitudinal (Saber ser)

Nombre del valor:
Ética

Área de Conocimiento (Saber)

Competencia(s)	
Comprende los principios avanzados de la Programación Orientada a Objetos, incluyendo el uso de clases abstractas, interfaces, miembros estáticos, serialización de objetos y programación multihilo, reconociendo su aplicación en el desarrollo de software mantenible y eficiente.	
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	
Implementa aplicaciones en Java utilizando clases abstractas, interfaces, miembros estáticos, serialización de objetos y programación multihilo para desarrollar soluciones modulares, reutilizables y con capacidad de procesamiento concurrente.	
Tipo de Actividad	Ponderación
CORTO 2 - Uso de Hilos para procesos concurrentes y serialización de datos persistentes.	5

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)	
Comprende el funcionamiento de los componentes avanzados de AWT y Swing, los administradores de diseño (Layout Managers) y el modelo de manejo de eventos para el desarrollo de interfaces gráficas de usuario dinámicas e interactivas.	
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 de usuario mediante la integración de componentes de AWT y Swing, administradores de diseño y mecanismos de manejo de eventos, desarrollando aplicaciones con una interacción intuitiva y una organización adecuada de la interfaz.	
Tipo de Actividad	Ponderación
Dashboard avanzado con Layout Managers y manejo de eventos de ratón.	0

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

Área Actitudinal (Saber ser)

Nombre del valor:
Curiosidad

Área de Conocimiento (Saber)

Competencia(s)	
Comprende los fundamentos del modelado de clases mediante UML y los conceptos esenciales de Cloud Computing, incluyendo sus modelos de servicio (SaaS, PaaS e IaaS) y principales proveedores, reconociendo su aplicación en el diseño e implementación de soluciones de software.	
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	
Evalúa el diseño de soluciones de software mediante la elaboración de diagramas de clases UML y la identificación del modelo de Cloud Computing más adecuado según los requerimientos de una aplicación, considerando criterios de organización, mantenibilidad y escalabilidad.	
Tipo de Actividad	Ponderación
Examen final	10

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

- Dean, J. S. y Dean, R. H. (2009). Introducción a la programación con Java. McGraw-Hill.

E-Grafía

- Manuales de Referencia de Java, <https://www.java.com/es/>