

LABORATORIO INTRODUCCIÓN A LA PROGRAMACIÓN Y COMPUTACIÓN 1

- SECCIÓN E

CÓDIGO:	770	PUNTEO NETO LABORATORIO:	20
ESCUELA DE INGENIERÍA EN:	CIENCIAS Y SISTEMAS	ÁREA A LA QUE PERTENECE:	DESARROLLO DE SOFTWARE
PRERREQUISITO:	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:	SEGUNDO 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 AUTOAPRENDIZAJE
Actividades en Clase	10	10
Proyectos	55	120
Prácticas	20	60
Tareas	5	10
Examen Final	10	1

Vigencia: Primer Semestre 2026

Página

1

TOTAL	100	201
-------	-----	-----

Equipo Académico

Coordinador del Área

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

Docente

Nombre del Docente Neftali de Jesus Calderon Mendez	Correo electrónico del Docente neftalicalderon@ingenieria.usac.edu.gt
---	---

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
--	-------	--------	-----------	--------	---------	--------

Día		X		X		
Horario						
Lugar						

Tutor(es)

Nombre del Tutor	Eduardo Isaí Ajsivinac Xico	
Correo electrónico institucional	3113272140409@ingenieria.usac.edu.gt	

Tipo		Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Clase	Día						X
	Horario						10:30-12:10
	Lugar						T3 - 114
Atención al Estudiante	Día	X	X	X	X	X	
	Horario	19:00-21:00	19:00-21:00	19:00-21:00	19:00-21:00	19:00-21:00	
	Lugar	Medios Oficiales	Medios Oficiales	Medios Oficiales	Medios Oficiales	Medios Oficiales	

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)	5
Diseño Didáctico	6
Sesión de Diagnóstico	6
Sesión No. 2, Unidad No. 1 - Introducción a los fundamentos de Programación	7
Sesión No. 3, Unidad No. 2 - Introducción al manejo de control de versiones	8
Sesión No. 4, Unidad No. 3 -Introducción a Java	9
Sesión No. 5, Unidad No. 4 - Manejo de Memoria	10
Sesión No. 6, Unidad No. 5 - Recursividad y Archivos	11
Sesión No. 7, Unidad No. 6 - Introducción a POO	12
Sesión No. 8, Unidad No. 7 - Introducción a Interfaces Gráficas y Patrones de Diseño	13
Sesión No. 9, Unidad No. 7 - POO Avanzado e Hilos	14
Sesión No. 10, Unidad No. 10 - Fundamentos de GUI	15
Sesión No. 11, Unidad No. 12 - UML y 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	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	Evalua 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
-------------------	-------------

Cuestionario en Google Forms	Se realizará un cuestionario en Google Forms para poder medir los conocimientos básicos del curso que el alumno posee previo al mismo.
------------------------------	--

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
Explicación y evaluación de herramientas y conocimientos	Se explicará a los estudiantes el uso de las herramientas a usar y se evaluará el conocimiento y uso de las mismas

Sesión No. 2, Unidad No. 1 - Introducción a los fundamentos de Programación

Área Actitudinal (Saber ser)

Nombre del valor: Puntualidad
Actitud de respetar y cumplir los horarios establecidos , llegando o actuando en el momento acordado, como muestra de responsabilidad y consideración hacia los demás.

Área de Conocimiento (Saber)

Competencia(s)	
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	
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	
Tema	Subtema
Introducción a los Fundamentos de Programación	Algoritmos
Introducción a los Fundamentos de Programación	Diagramas de Flujo

Área de Habilidades (Saber Hacer)

Competencia	
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	
Tipo de Actividad	Ponderación
Diseñar algoritmos aplicando diagramas de flujo y pseudocódigo para resolver problemas básicos de lógica computacional.	1 pts

Sesión No. 3, Unidad No. 2 - Introducción al versionamiento

Área Actitudinal (Saber ser)

Nombre del valor: Responsabilidad

Es la **capacidad de cumplir con los deberes y asumir las consecuencias de los propios actos**, actuando con compromiso, conciencia y respeto hacia los demás.

Área de Conocimiento (Saber)

Competencia(s)

Gestiona eficientemente el control de versiones de proyectos de software utilizando Git, aplicando buenas prácticas como el uso de repositorios, ramas, commits descriptivos, fusiones y resolución de conflictos, para mantener un historial claro y colaborativo del desarrollo.

Tema	Subtema
Introducción al manejo de control de versiones	Introducción a versionamiento
Introducción al manejo de control de versiones	Herramientas de control de versiones
Introducción al manejo de control de versiones	Tareas básicas de Git
Introducción al manejo de control de versiones	Colaboración en repositorios remotos

Área de Habilidades (Saber Hacer)

Competencia	
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	
Tipo de Actividad	Ponderación
Actividad planificada en UEDI	1 pts

Sesión No. 4, Unidad No. 3 - Introducción a Java

Área Actitudinal (Saber ser)

Nombre del valor: Disciplina
Capacidad de mantener orden, constancia y autocontrol para cumplir normas, hábitos u objetivos, incluso cuando no hay supervisión o motivación inmediata.

Área de Conocimiento (Saber)

Competencia(s)
En esta sesión, el estudiante conocerá los fundamentos del lenguaje Java, comprendiendo su propósito, características y componentes esenciales como variables, tipos de datos, operadores y estructuras básicas de entrada y salida. Se busca que el alumno adquiera una base sólida para el desarrollo posterior de programas orientados a objetos y el razonamiento lógico en la resolución de problemas.

Fomenta la curiosidad, la responsabilidad y la disposición al aprendizaje autónomo, al explorar un nuevo lenguaje de programación. Promueve la perseverancia ante los desafíos técnicos iniciales y el trabajo colaborativo en la construcción de conocimientos básicos. También impulsa la precisión y la atención al detalle, valores fundamentales en el desarrollo de software.

Tema	Subtema
Introducción a Java - Parte 1	¿Qué es Java?
Introducción a Java - Parte 2	Versiones y ambientes de java (JDK, JRE, JVM)
Introducción a Java - Parte 3	Características de Java
Introducción a Java - Parte 4	Comentarios de una línea y multilínea
Introducción a Java - Parte 5	Variables
Introducción a Java - Parte 6	Tipos primitivos y no primitivos
Introducción a Java - Parte 7	Casteos implícitos y explícitos

Área de Habilidades (Saber Hacer)

Competencia	
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	
Tipo de Actividad	Ponderación
Crear el repositorio IPC1_EJERCICIOS_carnet publico y realizar las instrucciones dadas. Al finalizar	1 pts
sube en un txt el link de tu repositorio.	

Sesión No. 5, Unidad No. 4 - Manejo de Memoria

Área Actitudinal (Saber ser)

Nombre del valor: Honestidad

Es la **actitud de actuar con verdad, sinceridad y coherencia moral**, siendo transparente con uno mismo y con los demás.

Área de Conocimiento (Saber)**Competencia(s)**

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

Evalúa 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

Tema	Subtema
Introducción a Java - Parte 4	Estructuras de Control
Introducción a Java - Parte 5	Ciclos
Introducción a Java - Parte 6	Try-catch-finally
Introducción a Java - Parte 7	Procedimientos y funciones
Introducción a Java - Parte 8	Debugging (Breakpoint [Inline, function], start, pause, continue, stop)
Manejo de Memoria	Arreglos y Matrices
Manejo de Memoria	Listas Dinámicas

Área de Habilidades (Saber Hacer)

Competencia	
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	
Tipo de Actividad	Ponderación
Actividad en UEDI	1 pts

Sesión No. 6, Unidad No. 5 - Recursividad y Archivos

Área Actitudinal (Saber ser)

Nombre del valor: Honestidad
La honestidad en esta sesión se aplica al reconocer y documentar la verdadera complejidad temporal de los algoritmos de ordenamiento ($O(n^2)$ para Bubble Sort) sin ocultar sus limitaciones, admitir cuando no se comprende completamente conceptos como recursividad o el Call Stack para buscar ayuda, reportar con transparencia los resultados reales de las pruebas incluyendo errores y casos de falla, no copiar código sin entenderlo o sin dar crédito a sus fuentes, y ser íntegro al manipular archivos de texto plano respetando la información procesada. Esta actitud permite un aprendizaje genuino, identificar áreas de mejora y construir soluciones confiables basadas en datos reales.

Área de Conocimiento (Saber)

Competencia(s)
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
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

Tema	Subtema
Manejo de Memoria	Métodos de Ordenamiento
Recursividad y archivos de texto	Pila de ejecución (Call Stack)
Recursividad y archivos de texto	Recursividad
Recursividad y archivos de texto	Manejo de archivos de texto plano

Área de Habilidades (Saber Hacer)

Competencia	
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	
Tipo de Actividad	Ponderación
Actividad en UEDI	1 pts

Sesión No. 7, Unidad No. 6- Introducción a POO

Área Actitudinal (Saber ser)

Nombre del valor: Honestidad
Es la virtud de decir la verdad y actuar con rectitud , sin engañar ni aprovecharse de los demás, manteniendo coherencia entre lo que se piensa, se dice y se hace.

Área de Conocimiento (Saber)

Competencia(s)

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

Evalua 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

Tema	Subtema
Conceptos y Pilares de POO	Concepto de abstracción y clasificación
Conceptos y Pilares de POO	Clases y objetos
Conceptos y Pilares de POO	Mensajes y métodos
Conceptos y Pilares de POO	Pilares de POO: Abstracción, Encapsulamiento, Polimorfismo, Herencia
Conceptos y Pilares de POO	Los miembros de una clase
Conceptos y Pilares de POO	Atributos y métodos (operaciones)
Conceptos y Pilares de POO	Constructores y Destructores
Conceptos y Pilares de POO	Modificadores de visibilidad: Privado, Público, Protegido

Área de Habilidades (Saber Hacer)

Competencia

Evalua 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

Tipo de Actividad	Ponderación
-------------------	-------------

Actividad en UEDI	1 pts
-------------------	-------

Sesión No. 8, Unidad No. 7 - Introducción a Interfaces Gráficas y patrones de Diseño

Área Actitudinal (Saber ser)

Nombre del valor: Disciplina
La disciplina es el hábito de hacer lo correcto de forma consistente, aunque sea difícil o poco atractivo en el momento, porque se comprende su valor a largo plazo.

Área de Conocimiento (Saber)

Competencia(s)	
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	
Evalúa 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	
Tema	Subtema
Introducción a Swing y AWT	Elementos Básicos de Swing
Patrón de diseño: MVC	Definición y sus componentes
Patrón de diseño: MVC	Modelo
Patrón de diseño: MVC	Vista
Patrón de diseño: MVC	Controlador

Área de Habilidades (Saber Hacer)

Competencia	
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	
Tipo de Actividad	Ponderación
Diseño de ventanas Swing, aplicando para el desarrollo del código el patrón de diseño MVC.	1 pts

Sesión No. 9, Unidad No. 9 - POO Avanzado e Hilos

Área Actitudinal (Saber ser)

Nombre del valor: Responsabilidad
Es la capacidad de cumplir compromisos y deberes , respondiendo por las propias acciones y sus consecuencias con seriedad y compromiso.

Área de Conocimiento (Saber)

Competencia(s)

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

Evalua 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

Tema	Subtema
Conceptos Avanzados de POO	Construcciones Abstractas: Clase Abstracta e Interface
Conceptos Avanzados de POO	Miembros estáticos (static) vs. miembros de instancia
Conceptos Avanzados de POO	Referencia "this"
Conceptos Avanzados de POO	Serialización de objetos en archivos
Manejo de Hilos en Java	Definición de Hilos
Manejo de Hilos en Java	Ciclo de vida de un hilo
Manejo de Hilos en Java	Creación de hilos
Manejo de Hilos en Java	Multihilos

Área de Habilidades (Saber Hacer)

Competencia	
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	
Tipo de Actividad	Ponderación
Creación de una aplicación en consola multihilo, que muestre cadenas de texto, aplicando conceptos del ciclo de vida de un hilo	1 pts

Sesión No. 10, Unidad No. 10 - Fundamentos de GUI

Área Actitudinal (Saber ser)

Nombre del valor: Responsabilidad

Durante la sesión, la responsabilidad se refleja en la disposición del estudiante para atender, participar activamente y aplicar los conceptos aprendidos en las actividades prácticas.

Área de Conocimiento (Saber)

Competencia(s)

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

Evalúa 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

Tema	Subtema
Fundamentos de GUI	Conceptos de GUI (Interfaz gráfica de usuario)
Fundamentos de GUI	¿Que es AWT?
Fundamentos de GUI	Componentes basicos de AWT
Fundamentos de GUI	Gestores de diseño (Layout Managers) en AWT
Fundamentos de GUI	Manejo de eventos en AWT
Fundamentos de GUI	¿Que es SWING?
Fundamentos de GUI	Componentes basicos de SWING
Fundamentos de GUI	Gestores de diseño (Layout Managers) en SWING
Fundamentos de GUI	Manejo de eventos en SWING

Fundamentos de GUI	Diferencias entre AWT y SWING
--------------------	-------------------------------

Área de Habilidades (Saber Hacer)

Competencia	
Diseña e implementa interfaces gráficas de usuario utilizando Java AWT y Swing, aplicando componentes, layouts y manejo de eventos para crear aplicaciones de escritorio funcionales, usables y mantenibles.	
Tipo de Actividad	Ponderación
Creación de una calculadora usando elementos Swing.	0.5 pts

Sesión No. 11, Unidad No. 12 - UML y Cloud Computing

Área Actitudinal (Saber ser)

Nombre del valor: Disciplina
Promueve la innovación, el pensamiento analítico y la adaptabilidad tecnológica, fomentando el trabajo colaborativo, la comunicación efectiva y la aplicación responsable de buenas prácticas de diseño en el desarrollo de sistemas e interfaces.

Área de Conocimiento (Saber)

Competencia(s)	
Evalúa 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	
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	
Tema	Subtema

Swing avanzado, UML y diagramas de clases	¿Qué es UML?
Swing avanzado, UML y diagramas de clases	Diagrama de clases
Swing avanzado, UML y diagramas de clases	Notaciones del diagrama de clases (clases, visibilidad, relaciones)
Cloud Computing	Ventajas y Desventajas
Cloud Computing	Tipos de Nube
Cloud Computing	Publica, privada, híbrida
Cloud Computing	Modelos en la Nube
Cloud Computing	SaaS, PaaS, IaaS
Cloud Computing	Servicios en la nube
Cloud Computing	Servicios en la nube

Área de Habilidades (Saber Hacer)

Competencia	
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	
Tipo de Actividad	Ponderación
Creación del modelo de datos de un sistema, usando diagramas de clases UML para representarlo.	0.5 pts

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

- OYANES, L. y ZAHONERO, I. "Programación en Java 2 (algoritmos, estructura de datos y programación orientada a objetos)". España, McGraw-Hill / Interamericana de España, S. A. 2002, PP 725
- JOYANES, L. "Programación en Turbo Pascal Versiones 5.5, 6.0, y 7.0", (2da Edición), México, McGrawHill / Interamericana de España, S. A. 1995, PP. 914
- Deitel & Deitel. "Cómo Programar en Java" (7ma Edición), México, Prentice Hall 2008, PP. 1280
- McLaughlin, B.; Pollice, G. y West, D. "Head First Object-Oriented Analysis & Design", EUA, O'Reilly Media 2006, PP. 636
- Freeman, E.; Robson, E.; Bates, B. y Sierra, K. "Head First Design Patterns", EUA, O'Reilly
- Mihaela Juganaru Mathieu, Introducción a la programación
- David Evans, Introduction to computing
- Jesus Fernandez-Pablo Guerron, David Zarruck, University of Pennsylvania
- Matloff Norman-Jay Peter, The art of Debugging
- OWASP Secure Coding Practices, Quick Reference Guide
- Media 2004, PP. 694

E-Grafía

- Manuales de Referencia de Java, <<http://www.sun.com/java>>
- Cualquier otro material (escrito o digital) entregado en clase.