

PROGRAMA DE LABORATORIO

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



INTRODUCCIÓN A LA PROGRAMACIÓN Y COMPUTACIÓN 1

CÓDIGO:	0770	PONDERACIÓN:	4
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:	SEGUNDO SEMESTRE 2025
HORAS POR SEMANA DEL CURSO:	4	HORAS POR SEMANA DEL LABORATORIO:	2
HORAS DE AUTOAPRENDIZAJE:	X	TOTAL DE HORAS DE APRENDIZAJE:	4
CATEDRÁTICO (A):	NEFTALI DE JESUS CALDERON MENDEZ	AUXILIAR:	MAX RODRIGO DURÁN CANTEO
EDIFICIO:	T3	SECCIÓN:	E
SALÓN DEL CURSO:	316	SALON DEL LABORATORIO:	113
DÍAS QUE SE IMPARTE EL CURSO:	Secciones B, C, D, E: Martes/Jueves Sección A: Sábado Sección F: Lunes/Miércoles	DÍAS QUE SE IMPARTE EL LABORATORIO:	Secciones A, B, C, D, E, F: Sábado
HORARIO DEL CURSO:	Secciones B,C,D,E 7:10 8:50 Sección A 7:10 – 10:30 Sección F 11:30 – 13:10	HORARIO DEL LABORATORIO:	Secciones A, B, C, D, E, F: 10:30 – 12:10

Breve 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.

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.....	6
Evaluación de conocimientos previos	6
Presentación del tutor	6
Presentación de los estudiantes.....	6
Presentación del programa del curso.....	6
Evaluación de conocimientos del laboratorio actual.....	6
Sesión No. 1, Unidad No. 1 - Introducción a la Programación	7
Valor de la semana (Saber ser).....	7
Conocimiento (Saber).....	7
Habilidades (Saber Hacer).....	7
Sesión No. 2, Unidad No. 1 - Introducción a la Programación	8
Valor de la semana (Saber ser).....	8
Conocimiento (Saber).....	8
Habilidades (Saber Hacer).....	8
Sesión No. 3, Unidad No. 2 - Introducción al versionamiento.....	9
Valor de la semana (Saber ser).....	9
Conocimiento (Saber).....	9
Habilidades (Saber Hacer).....	9
Sesión No. 4, Unidad No. 3 - Introducción a Java.....	10
Valor de la semana (Saber ser).....	10
Conocimiento (Saber).....	10
Habilidades (Saber Hacer).....	10
Sesión No. 5, Unidad No. 3 - Introducción a Java y Unidad No. 4 - Manejo de Memoria.....	11
Valor de la semana (Saber ser).....	11
Conocimiento (Saber).....	11
Habilidades (Saber Hacer).....	12
Sesión No. 6, Unidad No. 4 - Manejo de Memoria y Unidad No. 5 - Recursividad y Archivos ...	12
Valor de la semana (Saber ser).....	12

Conocimiento (Saber).....	12
Habilidades (Saber Hacer).....	13
Sesión No. 7, Unidad No. 6 - Introducción a POO	14
Valor de la semana (Saber ser).....	14
Conocimiento (Saber).....	14
Habilidades (Saber Hacer).....	15
Sesión No. 8, Unidad No. 7 - Introducción a Interfaces Gráficas y Patrones de diseño.....	15
Valor de la semana (Saber ser).....	15
Conocimiento (Saber).....	15
Habilidades (Saber Hacer).....	16
Sesión No. 9, Unidad No. 8 - POO Avanzado y Unidad No. 9 - Hilos	17
Valor de la semana (Saber ser).....	17
Conocimiento (Saber).....	17
Habilidades (Saber Hacer).....	17
Sesión No. 10, Unidad No. 10 - Fundamentos de GUI y Unidad No. 11. - Swing Avanzado y UML	18
Valor de la semana (Saber ser).....	18
Conocimiento (Saber).....	18
Habilidades (Saber Hacer).....	18
Sesión No. 11, Unidad No. 11 - Swing Avanzado y UML y Unidad No. 12 - Cloud computing.....	19
Valor de la semana (Saber ser).....	19
Conocimiento (Saber).....	19
Habilidades (Saber Hacer).....	20
Tiempo de Auto-aprendizaje	20
Rúbrica de Evaluación	20
Resumen de Ponderaciones	20
Normativa Académica y Ética del Curso	21
Equipo Académico.....	22
Coordinador del Área.....	22
Sección A.....	22
Sección B.....	23
Sección C.....	24
Sección D.....	25
Sección E.....	26
Sección F	27
Bibliografía	28

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.
4	Toma decisiones profesionales con base en fundamentos teóricos, datos e información pertinente, válida y confiable.
5	Comunica a diversos públicos, información de su campo profesional, en varios lenguajes y formatos de manera asertiva, clara, rigurosa y precisa, con el uso apropiado de recursos tecnológicos.

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 respetando y cuidando los recursos naturales, humanos y financieros.
4	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.
5	Construye soluciones integrales trabajando en forma colaborativa y propositiva en equipos

	interdisciplinarios, en forma presencial o utilizando plataformas virtuales.
--	--

Competencias del Laboratorio

Competencia(s) Específica(s)

No.	Competencia	Nivel de Aprendizaje
1	Diseña algoritmos aplicando diagramas de flujo y pseudocódigo para resolver problemas básicos de lógica computacional	Crear
2	Implementa programas simples utilizando lenguaje de programación estructurado (JAVA) conforme a especificaciones dadas	Aplicar
3	Analiza problemas computacionales empleando estrategias de resolución de problemas para optimizar soluciones algorítmicas	Evaluar
4	Aplica conceptos de programación mediante ejercicios prácticos y laboratorios respetando estándares básicos de calidad de software	Aplicar
5	Interpreta código fuente apoyándose en diagramas de flujo y pseudocódigo para detectar errores lógicos y de sintaxis	Comprender
6	Documenta programas desarrollados usando convenciones de codificación y comentarios estructurados para facilitar su comprensión y mantenimiento	Analizar

Competencia(s) General(es)

No.	Competencia	Nivel de Aprendizaje
1	Colabora en proyectos de programación utilizando plataformas de control de versiones básicas promoviendo el trabajo en equipo y la responsabilidad	Comprender

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 Conocimientos Previos	Un cuestionario con preguntas teóricas sobre conceptos como algoritmos y elementos básicos de programación.

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
Autoguía de instalación de herramientas iniciales	Guía con instrucciones claras en donde el estudiante prepara su entorno, instalando las herramientas específicas como JDK, Netbeans y Git, completando tareas sencillas y respondiendo un cuestionario asociado a cada ítem.

Sesión No. 1, Unidad No. 1 - Introducción a la Programación

Valor de la semana (Saber ser)

Nombre:
Puntualidad y Honestidad

Conocimiento (Saber)

Competencia(s)	
Diseñar algoritmos aplicando diagramas de flujo y pseudocódigo para resolver problemas básicos de lógica computacional	
Tema	Subtema
Introducción a los fundamentos de Programación	Algoritmos
Introducción a los fundamentos de Programación	Diagramas de Flujo

Habilidades (Saber Hacer)

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

Diseñar algoritmos aplicando diagramas de flujo y pseudocódigo para resolver problemas básicos de lógica computacional	Ejercicio	0
Analizar problemas computacionales empleando estrategias de resolución de problemas para optimizar soluciones algorítmicas	Actividad	0
-	Cuestionario /Corto	1

Sesión No. 2, Unidad No. 1 - Introducción a la Programación

Valor de la semana (Saber ser)

Nombre:
Honestidad

Conocimiento (Saber)

Competencia(s)	
Diseñar algoritmos aplicando diagramas de flujo y pseudocódigo para resolver problemas básicos de lógica computacional	
Tema	Subtema
Introducción a los fundamentos de Programación	Pseudocódigo

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Diseñar algoritmos aplicando diagramas de flujo y pseudocódigo para resolver problemas básicos de lógica computacional	Ejercicio	0
	Actividad	0
	Otros	1

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

Valor de la semana (Saber ser)

Nombre:
Honestidad

Conocimiento (Saber)

Competencia(s)	
Colaborar en proyectos de programación utilizando plataformas de control de versiones básicas promoviendo el trabajo en equipo y la responsabilidad	
Tema	Subtema
Introducción al manejo de control de versiones	Introducción a versionamiento
	Herramientas de control de versiones
	Tareas básicas de Git
	Colaboración en repositorios remotos

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Colaborar en proyectos de programación utilizando plataformas de control de versiones básicas promoviendo el trabajo en equipo y la responsabilidad	Ejercicio	0
	Actividad	0
	Otros	1

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

Valor de la semana (Saber ser)

Nombre:
Honestidad

Conocimiento (Saber)

Competencia(s)	
Aplicar conceptos de programación mediante ejercicios prácticos y laboratorios respetando estándares básicos de calidad 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
Introducción a Java - Parte 2	Operadores aritméticos, relacionales y lógicos
Introducción a Java - Parte 3	Input y output

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Aplicar conceptos de programación mediante ejercicios prácticos y laboratorios respetando estándares básicos de calidad de software	Ejercicio	0
	Actividad	0
	Corto/Cuestionario	1

Sesión No. 5, Unidad No. 3 - Introducción a Java y Unidad No. 4 - Manejo de Memoria

Valor de la semana (Saber ser)

Nombre:
Honestidad

Conocimiento (Saber)

Competencia	
Implementar programas simples utilizando lenguaje de programación estructurado (JAVA) conforme a especificaciones dadas	
Tema	Subtema
Introducción a Java - Parte 4	Estructuras de control (if, else if, else y switch)
Introducción a Java - Parte 5	Ciclos (for, while, do-while)

Competencia	
Aplicar conceptos de programación mediante ejercicios prácticos y laboratorios respetando estándares básicos de calidad de software	
Tema	Subtema
Introducción a Java - Parte 6	Try-catch-finally
Introducción a Java - Parte 7	Procedimientos y funciones

Competencia	
Interpretar código fuente apoyándose en diagramas de flujo y pseudocódigo para detectar errores lógicos y de sintaxis	
Tema	Subtema
Introducción a Java - Parte 8	Debugging (Breakpoint [Inline, function], start, pause, continue, stop)

Competencia	
Aplicar conceptos de programación mediante ejercicios prácticos y laboratorios respetando estándares básicos de calidad de software	
Tema	Subtema
Manejo de memoria	Arreglos y Matrices
Manejo de memoria	Listas Dinámicas

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Implementar programas simples utilizando lenguaje de programación estructurado (JAVA) conforme a especificaciones dadas	Ejercicio	0
	Actividad	0
	Corto/Cuestionario	1

Sesión No. 6, Unidad No. 4 - Manejo de Memoria y Unidad No. 5 - Recursividad y Archivos

Valor de la semana (Saber ser)

Nombre:
Honestidad

Conocimiento (Saber)

Competencia
Diseñar algoritmos aplicando diagramas de flujo y pseudocódigo para resolver problemas básicos de lógica computacional

Tema	Subtema
Manejo de memoria	Métodos de Ordenamiento

Competencia	
Analizar problemas computacionales empleando estrategias de resolución de problemas para optimizar soluciones algorítmicas	
Tema	Subtema
Recursividad y Archivos de texto plano	Pila de ejecución (Call Stack)

Competencia	
Diseñar algoritmos aplicando diagramas de flujo y pseudocódigo para resolver problemas básicos de lógica computacional	
Tema	Subtema
Recursividad y Archivos de texto plano	Recursividad

Competencia	
Aplicar conceptos de programación mediante ejercicios prácticos y laboratorios respetando estándares básicos de calidad de software	
Tema	Subtema
Recursividad y Archivos de texto plano	Manejo de archivos de texto plano

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Implementar programas simples utilizando lenguaje de programación estructurado (JAVA) conforme a especificaciones dadas	Ejercicio	0
Implementar programas simples utilizando lenguaje de programación estructurado (JAVA) conforme a	Actividad	0

especificaciones dadas		
Implementar programas simples utilizando lenguaje de programación estructurado (JAVA) conforme a especificaciones dadas	Corto/Cuestionario	1

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

Valor de la semana (Saber ser)

Nombre:
Honestidad

Conocimiento (Saber)

Competencia	
Aplicar conceptos de programación mediante ejercicios prácticos y laboratorios respetando estándares básicos de calidad de software	
Tema	Subtema
Conceptos y Pilares de POO	Concepto de abstracción y clasificación
	Clases y objetos
	Mensajes y métodos
	Pilares de POO: Abstracción, Encapsulamiento, Polimorfismo, Herencia
	Los miembros de una clase
	Atributos y métodos (operaciones)
	Constructores y Destructores
	Modificadores de visibilidad: Privado, Público, Protegido

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Aplicar conceptos de programación mediante	Ejercicio	0

ejercicios prácticos y laboratorios respetando estándares básicos de calidad de software		
Aplicar conceptos de programación mediante ejercicios prácticos y laboratorios respetando estándares básicos de calidad de software	Actividad	0
Aplicar conceptos de programación mediante ejercicios prácticos y laboratorios respetando estándares básicos de calidad de software	Corto/Cuestionario	1

Sesión No. 8, Unidad No. 7 - Introducción a Interfaces Gráficas y Patrones de diseño

Valor de la semana (Saber ser)

Nombre:
Honestidad

Conocimiento (Saber)

Competencia	
Implementar programas simples utilizando lenguaje de programación estructurado (JAVA) conforme a especificaciones dadas	
Tema	Subtema
Introducción a Swing y Awt	Elementos básicos de swing

Competencia	
Diseñar algoritmos aplicando diagramas de flujo y pseudocódigo para resolver problemas básicos de lógica computacional	
Tema	Subtema
Patrón de diseño: MVC	Definición y sus componentes

	Modelo
	Vista
	Controlador

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Implementar programas simples utilizando lenguaje de programación estructurado (JAVA) conforme a especificaciones dadas	Ejercicio	0
Implementar programas simples utilizando lenguaje de programación estructurado (JAVA) conforme a especificaciones dadas	Corto/Cuestionario	1

Sesión No. 9, Unidad No. 8 - POO Avanzado y Unidad No. 9 - Hilos

Valor de la semana (Saber ser)

Nombre:
Honestidad

Conocimiento (Saber)

Competencia	
Aplicar conceptos de programación mediante ejercicios prácticos y laboratorios respetando estándares básicos de calidad de software	
Tema	Subtema
Conceptos Avanzados de POO	Construcciones abstractas: Clase abstracta y Interface
	Miembros estáticos (static) vs. miembros de instancia
	Referencia "this"
	Serialización de objetos en archivos

Manejo de Hilos en Java	Definición de Hilos
	Ciclo de vida de un hilo
	Creación de hilos
	Multihilos

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Aplicar conceptos de programación mediante ejercicios prácticos y laboratorios respetando estándares básicos de calidad de software	Ejercicio	0
Aplicar conceptos de programación mediante ejercicios prácticos y laboratorios respetando estándares básicos de calidad de software	Otros	1

Sesión No. 10, Unidad No. 10 - Fundamentos de GUI y Unidad No. 11. - Swing Avanzado y UML

Valor de la semana (Saber ser)

Nombre:
Honestidad

Conocimiento (Saber)

Competencia	
Implementar programas simples utilizando lenguaje de programación estructurado (JAVA) conforme a especificaciones dadas	
Tema	Subtema
Fundamentos de GUI, AWT y SWING	Concepto de GUI (Interfaz gráfica de usuario).
Fundamentos de GUI, AWT y SWING	¿Que es AWT?
Fundamentos de GUI, AWT y SWING	Componentes basicos de AWT

Fundamentos de GUI, AWT y SWING	Gestores de diseño (Layout Managers) en AWT
Fundamentos de GUI, AWT y SWING	Manejo de eventos en AWT
Fundamentos de GUI, AWT y SWING	¿Que es SWING?
Fundamentos de GUI, AWT y SWING	Componentes basicos de SWING
SWING avanzado, UML y diagramas de clases	Gestores de diseño (Layout Managers) en SWING
SWING avanzado, UML y diagramas de clases	Manejo de eventos en SWING
SWING avanzado, UML y diagramas de clases	Diferencias entre AWT y SWING
SWING avanzado, UML y diagramas de clases	Drag and Drop

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Implementar programas simples utilizando lenguaje de programación estructurado (JAVA) conforme a especificaciones dadas	Ejercicio	0
Implementar programas simples utilizando lenguaje de programación estructurado (JAVA) conforme a especificaciones dadas	Actividad	0
Implementar programas simples utilizando lenguaje de programación estructurado (JAVA) conforme a especificaciones dadas	Otros	0.5

Sesión No. 11, Unidad No. 11 - Swing Avanzado y UML y Unidad No. 12 - Cloud computing

Valor de la semana (Saber ser)

Nombre:
Honestidad

Conocimiento (Saber)

Competencia	
Diseñar algoritmos aplicando diagramas de flujo y pseudocódigo para resolver problemas básicos de lógica computacional.	
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	Pública, privada, híbrida
Cloud Computing	Modelos en la Nube
Cloud Computing	SaaS, PaaS, IaaS
Cloud Computing	Servicios en la nube
Cloud Computing	Proveedores de Nube

Habilidades (Saber Hacer)

Competencia	Tipo de Actividad	Ponderación
Diseñar algoritmos aplicando diagramas de flujo y pseudocódigo para resolver problemas básicos de lógica computacional.	Ejercicio	0
Diseñar algoritmos aplicando diagramas de flujo y pseudocódigo para resolver problemas básicos de lógica computacional.	Actividad	0
Diseñar algoritmos aplicando diagramas de flujo y pseudocódigo para resolver problemas básicos de lógica computacional.	Corto/Cuestionario	0.5

Tiempo de Auto-aprendizaje

Tipo	Horas de Auto-aprendizaje
Proyectos	134
Prácticas	84
Tareas	0
Total	218

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	10
Proyectos	70
Prácticas	10
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 Francisco Orellana Lopez	Correo electrónico marlonorellana2005@gmail.com
--	---

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Día						X
Horario						7:10 - 10:30
Lugar						T-3, Salón 110

Tutor(es)

Nombre del Tutor	Anthony Alexander Aquino Santiago	
Correo electrónico institucional	3056670410301@ingenieria.usac.edu.gt	

Tipo		Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Clase	Día						X
	Horario						10:30 - 12:10
	Lugar						T-3, Salón 209
Atención al Estudiante	Día						
	Horario						

	Lugar						
--	-------	--	--	--	--	--	--

Sección B

Docente

Nombre del Docente William Estuardo Escobar Argueta	Correo electrónico
---	--------------------

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Día		X		X		
Horario		7:10 - 8:50		7:10 - 8:50		
Lugar		T3, 410		T3, 410		

Tutor(es)

Nombre del Tutor	Sebastián Alejandro Velásquez Bonilla	
Correo electrónico institucional	3180242840501@ingenieria.edu.gt	

Tipo		Lunes	Martes	Miércoles	Jueves	Vierne s	Sábado
Clase	Día						X
	Horario						10:30 - 12:10
	Lugar						T3, 312
Atención al Estudiante	Día						
	Horario						
	Lugar						

Sección C

Docente

Nombre del Docente Moisés Eduardo Velásquez Oliva	Correo electrónico 1924821300101@ingenieria.usac.edu.gt
---	---

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Día		X		X		
Horario		7:10 - 8:50		7:10 - 8:50		
Lugar		T-3, salón 414		T-3, salón 414		

Tutor(es)

Nombre del Tutor	Pablo José Oliva Bonilla	
Correo electrónico institucional	3006775020101@ingenieria.usac.edu.gt	

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

Sección D

Docente

Nombre del Docente Ing. Herman Igor Veliz Linares	Correo electrónico herman.veliz@gmail.com
--	--

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Día		X		X		
Horario		7:10-8:50		7:10-8:50		
Lugar		T-3, Salón 314		T-3, Salón 314		

Tutor(es)

Nombre del Tutor	Lesther Kevin Federico López Miculax	
Correo electrónico institucional	3010308470101@ingenieria.usac.edu.gt	

Tipo		Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Clase	Día						X
	Horario						10:30 - 12:10
	Lugar						T-3, Salón 111
Atención al Estudiante	Día						
	Horario						
	Lugar						

Sección E

Docente

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

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Día		X		X		
Horario		7:10 - 8:50		7:10 - 8:50		
Lugar		T-3, Salón 316		T-3, Salón 316		

Tutor(es)

Nombre del Tutor	Max Rodrigo Durán Canteo	
Correo electrónico institucional	2997082920101@ingenieria.usac.edu.gt	

Tipo		Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Clase	Día						X
	Horario						10:30 - 12:10
	Lugar						T-3, Salón 113
Atención al Estudiante	Día						
	Horario						
	Lugar						

Sección F

Docente

Nombre del Docente William Estuardo Escobar Argueta	Correo electrónico wescobara@ingenieria.usac.edu.gt
---	---

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Día	X		X			
Horario	11:30 - 13:10		11:30 - 13:10			
Lugar	T-3, Salón 412		T-3, Salón 412			

Tutor(es)

Nombre del Tutor	Zenaida Irazema Chacón García	
Correo electrónico institucional	3016759880101@ingenieria.usac.edu.gt	

Tipo		Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Clase	Día						X
	Horario						10:30 - 12:10
	Lugar						T-3, Salón 309
Atención al Estudiante	Día						
	Horario						
	Lugar						

Bibliografía

- JOYANES, 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
- BUDD, Timothy. "Introducción a la programación orientada a objetos", EUA, Addison, Wesley, Iberoamericana, S. A. 1994, P. 409
- 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
- Manuales de Referencia de Java, <<http://www.sun.com/java>>.