

PROGRAMA DE LABORATORIO

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

**LABORATORIO LENGUAJES FORMALES Y DE PROGRAMACIÓN - SECCIÓN B+**

CÓDIGO:	796	PUNTEO NETO LABORATORIO:	X
ESCUELA DE INGENIERÍA EN:	CIENCIAS Y SISTEMAS	ÁREA A LA QUE PERTENECE:	CIENCIAS DE LA COMPUTACIÓN
PRE REQUISITO:	770 - Introducción a la programación y computación 1 795 - Lógica de Sistemas 960 - Matemática Para Computación 1	POST REQUISITO:	777 - Organización de Lenguajes y Compiladores 1 772 - Estructuras de Datos 778 - Arquitectura de Computadoras y Ensambladores 1 773 - Manejo e Implementación de Archivos
CATEGORÍA:	OBLIGATORIO	VIGENCIA:	PRIMER SEMESTRE 2026

Descripción del Laboratorio

El laboratorio del curso de Lenguajes Formales de Programación busca introducir al estudiante con los fundamentos teóricos matemáticos y conceptos que fundamentan los lenguajes de programación. El estudiante debe de adquirir la base teórica necesaria para que pueda llevar a cabo el curso de lenguajes y compiladores. Se busca, además de definir diferentes modelos matemáticos asociados a la presentación de los diferentes tipos de lenguajes para luego implementar estos en los lenguajes de programación.

Resumen de Ponderaciones y Tiempo de Auto-aprendizaje

TIPO	PONDERACIÓN	HORAS DE AUTO-APRENDIZAJE
Actividades en Clase	6	1,25
Proyectos	65	55
Prácticas	15	10
Tareas	4	8
Examen Final	10	10
TOTAL	100	84,25

Equipo Académico

Coordinador del Área


Nombre: M.Sc. Luis Fernando Espino Barrios	Correo electrónico: usac.sistemas@gmail.com
---	--

Docente

Nombre del Docente DAVID ESTUARDO MORALES AJCOT	Correo electrónico del Docente
---	--------------------------------

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

Tutor(es)

Nombre del Tutor	Kewin Maslovy Patzan Tzun	
Correo electrónico institucional	3034573840109@ingenieria.usac.edu.gt	

Tipo		Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Clase	Día		X				
	Horario		9:00-10:40				
	Lugar		MEET				
Atención al Estudiante	Día						
	Horario						
	Lugar						

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 - Lenguajes de Programación.....	7
Sesión No. 3, Unidad No. 1 - Lenguajes de Programación.....	8
Sesión No. 4, Unidad No. 1 - Lenguajes de Programación.....	9
Sesión No. 5, Unidad No. 2 - Lenguajes Formales.....	10
Sesión No. 6, Unidad No. 2 - Lenguajes Formales.....	11
Sesión No. 7, Unidad No. 3 - Analisis Lexico.....	12
Sesión No. 8, Unidad No. 3 - Analisis Lexico.....	13
Sesión No. 9, Unidad No. 3 - Analisis Lexico.....	14
Sesión No. 10, Unidad No. 3 - Analisis Lexico y Unidad No. 4 - Análisis Sintáctico.....	15
Sesión No. 11, Unidad No. 4 - Análisis Sintáctico.....	16
Rúbrica de Evaluación.....	17
Normativa Académica y Ética del Curso.....	17

Competencias Vinculadas al Perfil del Egresado

Competencias Específicas

No.	Competencia
1	Demuestra pensamiento crítico, actitud investigativa y rigor analítico en el planteamiento y la resolución de problemas complejos.
2	Aplica los conocimientos de su disciplina en la elaboración, fundamentación y defensa de argumentos para prevenir y resolver problemas complejos en su campo profesional, identificando y aplicando innovaciones.
3	Se comunica correctamente en su lengua oficial y utiliza una lengua extranjera con el dominio requerido para el ejercicio de su profesión.

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 conocimientos tecnológicos con ética profesional respetando y cuidando los recursos naturales, humanos y financieros.
3	Actualiza permanentemente sus conocimientos relacionados con TIC en general, apoyándose en las estrategias de aprendizaje apropiadas.

Competencias del Laboratorio

Competencia(s) Específica(s)

No.	Competencia	Nivel de Aprendizaje
1	Analizar la evolución y clasificación de los lenguajes formales y de programación mediante la comparación de sus características, estructuras y herramientas para comprender su organización y funcionamiento	Analizar
2	Generar gramáticas libres de contexto mediante el análisis sintáctico para reconocer lenguajes formales.	Crear
3	Construir analizadores léxicos utilizando conocimientos de AFD, POO, recursividad y programación imperativa para definir cada estado del AFD	Crear
4	Construir analizadores sintácticos utilizando conocimientos de Gramáticas Libres del Contexto, POO, recursividad y programación imperativa para definir cada regla sintáctica	Crear

5	Comprender aspectos importantes sobre el lenguaje utilizado a través de su aplicación para implementar lo aprendido durante las clases teóricas	Comprender
6	Comprender el funcionamiento del lenguaje de programación a través de su aplicación para la resolución de problemas relacionados con las fases del compilador	Comprender

Competencia(s) General(es)

No.	Competencia	Nivel de Aprendizaje
1	El estudiante construye aplicaciones mediante el uso de análisis léxico y sintáctico utilizando herramientas de procesamiento de lenguajes formales.	Crear

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 de conocimientos previos.	Se realiza un cuestionario de 10 a 15 preguntas de opción múltiple o verdadero / falso sobre los temas fundamentales del curso. Para evaluar el conocimiento del estudiantado.

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
Cuestionario de conocimientos actuales del curso.	Se aplica una actividad interactiva mediante la plataforma Kahoot con 10 preguntas para evaluar el conocimiento previo del estudiante sobre el uso de herramientas específicas como JFLEX, CUP, editores de Código y conceptos básicos del análisis léxico y sintáctico.

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

Área Actitudinal (Saber ser)

Nombre del valor: Responsabilidad
La responsabilidad implica asumir con seriedad y compromiso cada tarea de aprendizaje, comprendiendo que los conocimientos adquiridos en clase serán la base para proyectos personales, académicos y profesionales.

Área de Conocimiento (Saber)

Competencia(s)	
Comprender el funcionamiento del lenguaje de programación a través de su aplicación para la resolución de problemas relacionados con las fases del compilador.	
Comprender aspectos importantes sobre el lenguaje utilizado a través de su aplicación para implementar lo aprendido durante las clases teóricas.	
Tema	Subtema
Introducción al lenguaje de Programación	Historia de la Programación
Introducción al lenguaje de Programación	Aspectos Básicos de la Programación
Introducción al lenguaje de Programación	Reglas de Sintaxis
Introducción al lenguaje de Programación	Estructura de un programa

Área de Habilidades (Saber Hacer)

Competencia	
Comprender el funcionamiento del lenguaje de programación a través de su aplicación para la resolución de problemas relacionados con las fases del compilador	
Tipo de Actividad	Ponderación
Cuestionario	0

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

Área Actitudinal (Saber ser)

Nombre del valor: Disciplina
La disciplina se refiere a la constancia y la capacidad de mantener un hábito de estudio ordenado y enfocado.

Área de Conocimiento (Saber)

Competencia(s)	
Analizar la evolución y clasificación de los lenguajes formales y de programación mediante la comparación de sus características, estructuras y herramientas para comprender su organización y funcionamiento.	
Tema	Subtema
Buenas Prácticas y funciones en Programación	Convenciones de nombres y estilo de código
Buenas Prácticas y funciones en Programación	Gestión de memoria y optimización
Buenas Prácticas y funciones en Programación	Definición y uso de métodos y funciones
Buenas Prácticas y funciones en Programación	Creación y uso de clases
Buenas Prácticas y funciones en Programación	Diferencias entre métodos y funciones

Área de Habilidades (Saber Hacer)

Competencia	
Comprender el funcionamiento del lenguaje de programación a través de su aplicación para la resolución de problemas relacionados con las fases del compilador	
Tipo de Actividad	Ponderación
Actividades	0

Sesión No. 4, Unidad No. 1 - Lenguajes de Programación

Área Actitudinal (Saber ser) Lenguajes de Programación

Nombre del valor: Perseverancia
Es la capacidad de mantener el esfuerzo y la determinación para alcanzar una meta a pesar de los obstáculos, las dificultades o los fracasos que puedan surgir en el camino.

Área de Conocimiento (Saber)

Competencia(s)	
Comprender el funcionamiento del lenguaje de programación a través de su aplicación para la resolución de problemas relacionados con las fases del compilador.	
Analizar la evolución y clasificación de los lenguajes formales y de programación mediante la comparación de sus características, estructuras y herramientas para comprender su organización y funcionamiento.	
Tema	Subtema
Estructuras de Datos y Control en Programación	Arreglos en C++
Estructuras de Datos y Control en Programación	Diccionarios y estructuras de Datos
Estructuras de Datos y Control en Programación	Iteración en C++
Estructuras de Datos y Control en Programación	Lectura de Archivos
Estructuras de Datos y Control en Programación	Escritura de Archivos

Área de Habilidades (Saber Hacer)

Competencia	
Analizar la evolución y clasificación de los lenguajes formales y de programación mediante la comparación de sus características, estructuras y herramientas para comprender su organización y funcionamiento.	
Tipo de Actividad	Ponderación
Cuestionario	2

Sesión No. 5, Unidad No. 2 - Lenguajes Formales

Área Actitudinal (Saber ser)

Nombre del valor: Pensamiento Crítico
Es la capacidad de analizar, evaluar y cuestionar la información de manera lógica, objetiva y racional para formarse un juicio o tomar decisiones fundamentadas. Implica no aceptar ideas o argumentos sin examinarlos cuidadosamente, considerando la evidencia, los supuestos, las implicaciones y posibles sesgos.

Área de Conocimiento (Saber)

Competencia(s)	
Comprender el funcionamiento del lenguaje de programación a través de su aplicación para la resolución de problemas relacionados con las fases del compilador .	
Analizar la evolución y clasificación de los lenguajes formales y de programación mediante la comparación de sus características, estructuras y herramientas para comprender su organización y funcionamiento.	
Tema	Subtema
Lenguajes	Lenguajes Natural
Lenguajes	Lenguajes Formales
Lenguajes	Lenguajes de Programación
Lenguajes	Evolución de los Lenguajes de Programación
Lenguajes	Paradigmas
Lenguajes	Generaciones de lenguajes de programación

Área de Habilidades (Saber Hacer)

Competencia	
Comprender el funcionamiento del lenguaje de programación a través de su aplicación para la resolución de problemas relacionados con las fases del compilador	
Tipo de Actividad	Ponderación
Actividades	0

Sesión No. 6, Unidad No. 2 - Lenguajes Formales

Área Actitudinal (Saber ser)

Nombre del valor: Paciencia
Es la capacidad de mantener la calma, la serenidad y el control emocional ante situaciones difíciles, retrasos, frustraciones o cuando los resultados deseados tardan en llegar.

Área de Conocimiento (Saber)

Competencia(s)	
Comprender el funcionamiento del lenguaje de programación a través de su aplicación para la resolución de problemas relacionados con las fases del compilador	
Construir analizadores léxicos utilizando conocimientos de AFD, POO, recursividad y programación imperativa para definir cada estado del AFD	
Tema	Subtema
Procesadores de Lenguaje	Intérprete
Procesadores de Lenguaje	Compilador
Procesadores de Lenguaje	Estructura de un compilador
Procesadores de Lenguaje	Diferencias y Ejemplos
Procesadores de Lenguaje	Herramientas
Procesadores de Lenguaje	Jerarquía de Chomsky
Procesadores de Lenguaje	Clasificación de Gramáticas

Área de Habilidades (Saber Hacer)

Competencia	
Construir analizadores léxicos utilizando conocimientos de AFD, POO, recursividad y programación imperativa para definir cada estado del AFD	
Tipo de Actividad	Ponderación
Ejercicio práctico	0

Sesión No. 7, Unidad No. 3 - Analisis Lexico

Área Actitudinal (Saber ser)

Nombre del valor: Autoconocimiento
Permite al estudiante identificar sus fortalezas, intereses y aspectos personales relevantes.

Área de Conocimiento (Saber)

Competencia(s)	
Analizar la evolución y clasificación de los lenguajes formales y de programación mediante la comparación de sus características, estructuras y herramientas para comprender su organización y funcionamiento.	
Construir analizadores léxicos utilizando conocimientos de AFD, POO, recursividad y programación imperativa para definir cada estado del AFD.	
Tema	Subtema
Definición y Función del analizador léxico	Patrones
Definición y Función del analizador léxico	Tokens
Definición y Función del analizador léxico	Lexemas
Definición y Función del analizador léxico	Errores Léxicos
Definición y Función del analizador léxico	Ejemplos de tokens en diferentes lenguajes
Operaciones Entre Lenguajes y Expresiones regulares	Interoperabilidad entre Lenguajes
Operaciones Entre Lenguajes y Expresiones regulares	Concepto de interoperabilidad
Operaciones Entre Lenguajes y Expresiones regulares	Traducción y Compilación: Proceso de traducción de un lenguaje a otro
Operaciones Entre Lenguajes y Expresiones regulares	Buenas Prácticas en Interoperabilidad
Operaciones Entre Lenguajes y Expresiones regulares	¿Qué son las expresiones regulares?
Operaciones Entre Lenguajes y Expresiones regulares	Aplicaciones de Expresiones Regulares en Análisis Léxico
Operaciones Entre Lenguajes y Expresiones regulares	Diagrama de transición de estados
Operaciones Entre Lenguajes y Expresiones regulares	Tablas de Transición

Área de Habilidades (Saber Hacer)

Competencia	
Construir analizadores léxicos utilizando conocimientos de AFD, POO, recursividad y programación imperativa para definir cada estado del AFD	
Tipo de Actividad	Ponderación
Ejercicio práctico	2

Sesión No. 8, Unidad No. 3 - Analisis Lexico

Área Actitudinal (Saber ser)

Nombre del valor: Curiosidad
Se estimula al presentarse contenidos nuevos que despiertan interés por aprender.

Área de Conocimiento (Saber)

Competencia(s)	
Construir analizadores léxicos utilizando conocimientos de AFD, POO, recursividad y programación imperativa para definir cada estado del AFD.	
Analizar la evolución y clasificación de los lenguajes formales y de programación mediante la comparación de sus características, estructuras y herramientas para comprender su organización y funcionamiento.	
Tema	Subtema
Automata Finitos	Definición de AFN
Automata Finitos	Estructura de un AFN
Automata Finitos	Ejemplo de AFN
Automata Finitos	Definición de AFD
Automata Finitos	Estructura de un AFD
Automata Finitos	Ejemplo de AFD

Área de Habilidades (Saber Hacer)

Competencia	
Construir analizadores léxicos utilizando conocimientos de AFD, POO, recursividad y programación imperativa para definir cada estado del AFD	
Tipo de Actividad	Ponderación
Ejercicio práctico	0

Sesión No. 9, Unidad No. 3 - Analisis Lexico

Área Actitudinal (Saber ser)

Nombre del valor: Compromiso
Se impulsa al conocer y aceptar activamente el programa del curso.

Área de Conocimiento (Saber)

Competencia(s)	
Construir analizadores léxicos utilizando conocimientos de AFD, POO, recursividad y programación imperativa para definir cada estado del AFD	
Generar gramáticas libres de contexto mediante el análisis sintáctico para reconocer lenguajes formales.	
Tema	Subtema
Conversion AFN a AFD	Método del Árbol
Conversion AFN a AFD	Anulables
Conversion AFN a AFD	Cálculo de First, Last y Next

Área de Habilidades (Saber Hacer)

Competencia	
Construir analizadores léxicos utilizando conocimientos de AFD, POO, recursividad y programación imperativa para definir cada estado del AFD	
Tipo de Actividad	Ponderación
Ejercicio practico	0

Sesión No. 10, Unidad No. 3 - Analisis Lexico y Unidad No. 4 - Análisis Sintáctico

Área Actitudinal (Saber ser)

Nombre del valor: Perseverancia
Aborda problemas complejos como los autómatas de pila sin rendirse.

Área de Conocimiento (Saber)

Competencia(s)	
Construir analizadores sintácticos utilizando conocimientos de Gramáticas Libres del Contexto, POO, recursividad y programación imperativa para definir cada regla sintáctica	
Tema	Subtema
Optimizacion de Estados	Minimización de un AFD
Optimizacion de Estados	Identificación de Estados Equivalentes
Optimizacion de Estados	Representación Compacta de AFD
Optimizacion de Estados	Ventajas de una representación optimizada
Funcion del Analizador Sintactico	Definición de análisis sintáctico
Funcion del Analizador Sintactico	Propósito del analizador sintáctico en un compilador
Funcion del Analizador Sintactico	Diferencias entre análisis léxico y sintáctico
Funcion del Analizador Sintactico	Analizadores sintácticos ascendentes
Funcion del Analizador Sintactico	Analizadores sintácticos descendentes
Funcion del Analizador Sintactico	Comparación de técnicas de análisis sintáctico
Lenguajes Libres de Contexto	Definición de lenguajes libres de contexto (LFC)
Lenguajes Libres de Contexto	Características de LFC
Lenguajes Libres de Contexto	Importancia de los LFC en la teoría de lenguajes de programación
Lenguajes Libres de Contexto	Gramáticas Tipo 2
Lenguajes Libres de Contexto	Ejemplos de generación de cadenas usando gramáticas Tipo 2

Lenguajes Libres de Contexto	Aplicaciones prácticas de gramáticas Tipo 2 en lenguajes de programación
------------------------------	--

Área de Habilidades (Saber Hacer)

Competencia	
Comprender el funcionamiento del lenguaje de programación a través de su aplicación para la resolución de problemas relacionados con las fases del compilador	
Tipo de Actividad	Ponderación
Cuestionario	2

Sesión No. 11, Unidad No. 4 - Análisis Sintáctico

Área Actitudinal (Saber ser)

Nombre del valor: Autonomía
Desarrolla aplicaciones de análisis sintáctico de forma independiente.

Área de Conocimiento (Saber)

Competencia(s)	
El estudiante construye aplicaciones mediante el uso de analisis lexico y sintactico utilizando herramientas de procesamiento de lenguajes formales.	
Construir analizadores sintácticos utilizando conocimientos de Gramáticas Libres del Contexto, POO, recursividad y programación imperativa para definir cada regla sintáctica .	
Tema	Subtema
Árboles de Derivación y Ambigüedad	Árboles de derivación
Árboles de Derivación y Ambigüedad	Definición y construcción de árboles de derivación
Árboles de Derivación y Ambigüedad	Ambigüedad
Árboles de Derivación y Ambigüedad	Concepto de ambigüedad en gramáticas
Árboles de Derivación y Ambigüedad	Ejemplos de gramáticas ambiguas
Árboles de Derivación y Ambigüedad	Recursividad
Árboles de Derivación y Ambigüedad	Definición de recursividad en gramáticas
Árboles de Derivación y Ambigüedad	Ejemplos de gramáticas recursivas
Autómatas de Pila	Introducción a los autómatas de pila

Autómatas de Pila	Definición y funcionamiento de un autómata de pila
Autómatas de Pila	Diferencias entre autómatas de pila y autómatas finitos
Autómatas de Pila	Procesamiento
Autómatas de Pila	Tipos de aceptación
Autómatas de Pila	Estado Final
Autómatas de Pila	Pila vacía
Autómatas de Pila	Autómata de Pila desde Gramática Tipo 2

Área de Habilidades (Saber Hacer)

Competencia	
Generar gramáticas libres de contexto mediante el análisis sintáctico para reconocer lenguajes formales.	
Tipo de Actividad	Ponderación
Ejercicio práctico	0

Calendarización de Práctica/Proyectos

Actividad	Publicación	Entrega
Práctica	10/02/2026	01/03/2026
Proyecto 1	03/03/2026	31/03/2026
Proyecto 2	31/03/2026	30/04/2026

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.