

SÍLABO



FACULTAD DE INGENIERÍA

MARZO 2024-AGOSTO 2024

NOMBRE DE LA ASIGNATURA		CÓDIGO:	18585
LENGUAJES DE PROGRAMACIÓN - GRUPO: 1			
CARRERA	COMPUTACION REDISEÑO		
CICLO O SEMESTRE	QUINTO NIVEL	EJE DE FORMACIÓN	PROFESIONALES, PRAXIS PROFESIONAL
CRÉDITOS DE LA ASIGNATURA	4	MODALIDAD:	PRESENCIAL

CARGA HORARIA

COMPONENTES DEL APRENDIZAJE	Horas / Semana	Horas / Periodo Académico
APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	4.0	64.0
APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	2.0	32.0
APRENDIZAJE AUTÓNOMO (AA)	6.0	96.0
Total Horas:	12.0	192.0

PROFESOR(ES) RESPONSABLE(S):

MEJIA PESANTEZ PIEDAD MAGALI - (P.M.)	(magali.mejia@ucuenca.edu.ec)	PRINCIPAL
---------------------------------------	---------------------------------	-----------

DESCRIPCIÓN DE LA ASIGNATURA:

Resumen descriptivo en torno al propósito, la estrategia metodológica y el contenido fundamental de la asignatura.

Esta asignatura permitirá que el estudiante seleccione un lenguaje de programación de acuerdo al problema que necesita resolver para lo cual conocerá las características de cada uno de los paradigmas de los lenguajes de programación (LP).

REQUISITOS DE LA ASIGNATURA

PRE-REQUISITOS		CO-REQUISITOS	
Asignatura	Código	Asignatura	Código
PROGRAMACIÓN III (ESTRUCTURA DE ARCHIVOS)	18566	BASE DE DATOS I: DISEÑO Y PRINCIPIOS	18583

OBJETIVO(S) DE LA ASIGNATURA:

Objetivos general y específicos de la asignatura en relación al Perfil de salida de la carrera.

Objetivo general: Implementar aplicaciones utilizando el lenguaje de programación adecuado de acuerdo al problema que se debe resolver, a través de los paradigmas de programación Orientado a Objetos, Funcional y Lógico.

Objetivos específicos:

- Proveer al estudiante de los conceptos teóricos que fundamentan los lenguajes de programación mediante una comparación entre los diferentes paradigmas de programación para adiestrar al estudiante en la utilización de lenguajes de programación Python, PROLOG y SCALA.
- Estudiar las características de los paradigmas de los lenguajes de programación
- Comprender la sintaxis y semántica de los LP
- Comparar los tipos de datos, estructuras y otros de lenguajes de programación como C, C++ y Java

5. Revisar los conceptos básicos de la Programación Orientada a Objetos utilizando Java
6. Utilizar el lenguaje de programación multiparadigma Python (imperativo, orientado a objetos y funcional)
7. Utilizar el lenguaje de programación lógica PROLOG
8. Utilizar el lenguaje de programación funcional y orientado a objetos SCALA, que permite interoperabilidad con Java

LOGRO DE LOS RESULTADOS DE APRENDIZAJE, INDICADOR(ES) Y ESTRATEGIA(S) DE EVALUACIÓN

Resultados o Logros de Aprendizaje (RdA's) de la Unidad de Organización Curricular (UOC) correspondiente, Indicadores y Estrategias de Evaluación de la Asignatura, tomando como referencia el Perfil de salida (PdS) y la Organización Curricular (OC) del Proyecto de Carrera (PdC).

RESULTADOS O LOGROS DE APRENDIZAJE	INDICADORES	ESTRATEGIAS DE EVALUACIÓN
RdA1. Comprende los diferentes paradigmas de los lenguajes de la programación y reconoce la genealogía de los lenguajes más populares	<ul style="list-style-type: none"> Define con claridad las características de los diferentes paradigmas de programación Identifica la genealogía de los lenguajes de programación 	<ul style="list-style-type: none"> ELABORA UNA LÍNEA DE TIEMPO CON LOS LENGUAJES DE PROGRAMACIÓN MÁS UTILIZADOS TRABAJO COLABORATIVO SOBRE LOS PARADIGMAS DE PROGRAMACIÓN
RdA2. Identifica los conceptos básicos del léxico, sintaxis y semántica de los LP y utiliza los métodos de representación	<ul style="list-style-type: none"> Utiliza el método EBNF para describir la sintaxis de un LP Dibuja el árbol sintáctico de una expresión 	<ul style="list-style-type: none"> EJERCICIOS SOBRE ÁRBOLES SINTÁCTICOS Y EBNF
RdA3. Compara los Tipos de datos, ámbito, estructuras de control, expresiones y asignación de los lenguajes más populares	<ul style="list-style-type: none"> Explica los tipos de datos, el ámbito, estructuras de control, expresiones y asignaciones de varios lenguajes 	<ul style="list-style-type: none"> CONTESTA UN CUESTIONARIO DE LA PLATAFORMA VIRTUAL
RdA4. Compara las Iteraciones y subprogramas en diferentes LP	<ul style="list-style-type: none"> Explica las Iteraciones y subprogramas en diferentes LP 	<ul style="list-style-type: none"> RESUELVE UN CUESTIONARIO SOBRE LA UNIDAD QUE SE ENCUENTRA EN LA PLATAFORMA VIRTUAL
RdA5. Aplica correctamente las características de un lenguaje de Programación Orientado a Objetos	<ul style="list-style-type: none"> Utiliza correctamente los lenguajes de programación orientados a objetos 	<ul style="list-style-type: none"> DESARROLLA UNA APLICACIÓN DE PYTHON ORIENTADA A OBJETOS Y PROGRAMACIÓN FUNCIONAL EN CONSOLA DESARROLLA UNA APLICACIÓN EN PYTHON CON INTERFACES GRÁFICAS DE USUARIO
RdA6. Aplica correctamente las características de un lenguaje funcional	<ul style="list-style-type: none"> Utiliza correctamente los lenguajes funcionales 	<ul style="list-style-type: none"> REALIZA UNA APLICACIÓN EN SCALA UTILIZANDO PROGRAMACIÓN FUNCIONAL Y ORIENTADA A OBJETOS
RdA7. Aplica correctamente las características de un lenguaje lógico	<ul style="list-style-type: none"> Utiliza correctamente los lenguajes lógicos 	<ul style="list-style-type: none"> RESUELVE PROBLEMAS UTILIZANDO EL LENGUAJE DE PROGRAMACIÓN LÓGICA PROLOG

CONTENIDOS, SESIONES Y ACTIVIDADES DE APRENDIZAJE

Título de la Unidad, sub -unidades, nro. de sesión y actividades para los componentes de aprendizaje.

SUB-UNIDADES	Nro. SESIÓN	COMPONENTE DE APRENDIZAJE	ACTIVIDADES DE APRENDIZAJE
1. INTRODUCCIÓN Y CONCEPTOS GENERALES			

SUB-UNIDADES	Nro. SESIÓN	COMPONENTE DE APRENDIZAJE	ACTIVIDADES DE APRENDIZAJE	
1. Razones para estudiar lenguajes de programación	1	APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	EXPLICACIÓN DEL SÍLABO DE LA ASIGNATURA	2 horas
2. Dominios de programación			Clase magistral	3 horas
3. Criterio de evaluación de los LP			EXPOSICION DE LOS ESTUDIANTES	2 horas
4. Factores para el desarrollo de LP			PRUEBA DE LA UNIDAD	1 horas
5. Categorías de LP		APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	TALLER 1 DIAGNOSTICO SOBRE UN PROBLEMA DE PROGRAMACIÓN EN LENGUAJES C Y JAVA	2 horas
6. Métodos de implementación				
7. Ambientes de programación		APRENDIZAJE AUTÓNOMO (AA)	ELABORA UNA LÍNEA DE TIEMPO CON LOS LENGUAJES DE PROGRAMACIÓN MÁS UTILIZADOS	4 horas
8. Genealogía de los LP más usados			TRABAJO COLABORATIVO SOBRE LOS PARADIGMAS DE PROGRAMACIÓN	4 horas
2. ANÁLISIS LÉXICO, SINTÁCTICO Y SEMÁNTICO				
1. Introducción	2	APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	CLASES MAGISTRALES	2 horas
2. Análisis léxico			EJERCICIOS SOBRE EL ÁRBOL SINTÁCTICO	1 horas
3. El analizador sintáctico			EJERCICIOS SOBRE EBNF	1 horas
4. Sintaxis		APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	TALLER 2 SOBRE EL ARBOL SINTÁCTICO	2 horas
5. Semántica			TALLER 3 SOBRE GRAFOS SINTÁCTICOS	2 horas
		APRENDIZAJE AUTÓNOMO (AA)	EJERCICIOS SOBRE ÁRBOLES SINTÁCTICOS Y EBNF	2 horas
3. TIPOS DE DATOS, ÁMBITO, ESTRUCTURAS DE CONTROL, EXPRESIONES Y ASIGNACIÓN				
1. Evolución de la abstracción de datos y de control	3	APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	CLASES MAGISTRALES	4 horas
2. Nombres, variables			RESOLUCIÓN DE EJERCICIOS	2 horas
3. El concepto de ligaduras		APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	TALLER 4 SOBRE EJERCICIOS DE PROGRAMACIÓN SOBRE EL CAPÍTULO	2 horas
4. Tipos de datos				
5. Ámbito y tiempo de vida		APRENDIZAJE AUTÓNOMO (AA)	DESARROLLA UN CUESTIONARIO SOBRE EL CAPÍTULO	2 horas
6. Ambientes de referencia, constantes nominales				
4. ITERACIONES Y SUBPROGRAMAS				
1. Estructuras de control a nivel de unidad	4	APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	CLASE MAGISTRAL	2 horas
2. Estructuras de control a nivel de sentencia			PRUEBA UNIDAD 2, 3 Y 4	2 horas
3. Subprogramas		APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	TALLER 5 EJERCICIOS SOBRE PROCEDIMIENTOS	2 horas
	APRENDIZAJE AUTÓNOMO (AA)	DESARROLLA UN CUESTIONARIO SOBRE EL CAPÍTULO	2 horas	
5. PROGRAMACIÓN ORIENTADA A OBJETOS CON PYTHON				

SUB-UNIDADES	Nro. SESIÓN	COMPONENTE DE APRENDIZAJE	ACTIVIDADES DE APRENDIZAJE	
1. Introducción a TDA y OO 2. Conceptos básicos 3. Clases y Objetos 4. Herencia 5. Manejo de excepciones 6. Lenguaje python	5	APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	CLASE MAGISTRAL	2 horas
			EXPOSICIÓN DE ESTUDIANTES SOBRE PYTHON	2 horas
			CLASES MAGISTRALES PYTHON ORIENTADO A OBJETOS, FUNCIONAL Y EL DISEÑADOR DE INTERFACES GRÁFICAS	16 horas
			PRUEBA DE LA UNIDAD	2 horas
		APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	TALLER SOBRE PYTHON: PROGRAMACIÓN ORIENTADA A OBJETOS, PROGRAMACIÓN FUNCIONAL E INTERFACES GRÁFICAS	8 horas
		APRENDIZAJE AUTÓNOMO (AA)	REALIZA UNA APLICACIÓN DE PYTHON ORIENTADA A OBJETOS Y PROGRAMACIÓN FUNCIONAL EN CONSOLA	6 horas
REALIZA UNA APLICACIÓN EN PYTHON CON INTERFACES GRÁFICAS DE USUARIO	30 horas			
6. PROGRAMACIÓN FUNCIONAL				
1. Introducción 2. Lenguaje SCALA	6	APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	CLASES MAGISTRALES SOBRE	10 horas
			PRUEBA DE LA UNIDAD	2 horas
		APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	TALLER SOBRE SCALA: PROGRAMACIÓN ORIENTADA A OBJETOS, FUNCIONAL E INTERFACES GRÁFICAS CON JAVA	8 horas
		APRENDIZAJE AUTÓNOMO (AA)	REALIZA UNA APLICACIÓN EN SCALA UTILIZANDO PROGRAMACIÓN FUNCIONAL Y ORIENTADA A OBJETOS CON INTERFACES GRÁFICAS EN JAVA	40 horas
7. PROGRAMACIÓN LÓGICA				
1. Introducción 2. Cláusulas de Horn 3. El lenguaje PROLOG	7	APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	CLASES MAGISTRALES SOBRE PROLOG	6 horas
			PRUEBA DE LA UNIDAD	2 horas
		APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	TALLER SOBRE PROLOG	6 horas
		APRENDIZAJE AUTÓNOMO (AA)	RESUELVE PROBLEMAS UTILIZANDO EL LENGUAJE DE PROGRAMACIÓN LÓGICA PROLOG	6 horas
		APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	64 horas	
		APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	32 horas	
APRENDIZAJE AUTÓNOMO (AA)	96 horas			
		Total Planificación:	192 horas	

RECURSOS O MEDIOS PARA EL APRENDIZAJE

Equipos, materiales, instrumentos tecnológicos, reactivos, entre otros, que serán utilizados durante el desarrollo de la asignatura.

- CURSO VIRTUAL MOODLE CON LIBROS INTERACTIVOS, PRESENTACIONES UTILIZANDO EL PROYECTOR MULTIMEDIA

CRITERIOS PARA LA ACREDITACIÓN DE LA ASIGNATURA

Parámetros de acreditación, tomando como referencia los Resultados de Aprendizaje (RdA's), indicadores y criterios de evaluación planteados y en base a la normativa de evaluación y calificaciones vigente en la Universidad de Cuenca y Consejo de Educación Superior (CES).

CRITERIO GENERAL DE ACREDITACIÓN	PUNTAJE
PRUEBAS	20
TRABAJOS	20
TALLERES	10
EXAMENES	50
TOTAL:	100

	DETALLE DE CRITERIOS DE ACREDITACIÓN	PUNTAJE / CRITERIO GENERAL	
C94	APROVECHAMIENTO I		
	PRUEBA DE LAS UNIDADES 2, 3 Y 4 CONCEPTOS FUNDAMENTALES LENGUAJES DE PROGRAMACIÓN	5	PRUEBAS
	PRUEBA DE UNIDAD 1: INTRODUCCIÓN A LOS LENGUAJES DE PROGRAMACIÓN	4	PRUEBAS
	LÍNEA DE TIEMPO DE LOS LENGUAJES DE PROGRAMACIÓN MÁS UTILIZADOS	3	TRABAJO
	TRABAJO COLABORATIVO SOBRE LOS PARADIGMAS DE PROGRAMACIÓN	3	TRABAJO
	TRABAJO COLABORATIVO SOBRE PYTHON	3	TRABAJO
	CUESTIONARIO SOBRE LA UNIDAD 2	3	TRABAJO
	TALLERES DESARROLLADOS EN EL COMPONENTE APE DESDE EL CAP1 HASTA EL 4	5	TALLERES
C95	INTERCICLO		
	EXAMEN INTERCICLO	20	EXAMENES
C96	APROVECHAMIENTO II		
	PRUEBA DE LA UNIDAD 5 PROGRAMACIÓN ORIENTADA A OBJETOS PYTHON	4	PRUEBAS
	PRUEBA DE LA UNIDAD 7 PROGRAMACIÓN LÓGICA PROLOG	3	PRUEBAS
	PRUEBA DE LA UNIDAD 6 PROGRAMACIÓN FUNCIONAL Y POO EN SCALA	4	PRUEBAS
	EJERCICIOS UTILIZANDO PROLOG	3	TRABAJO
	TRABAJO EN PYTHON UTILIZANDO INTERFACES GRÁFICAS DE USUARIO	5	TRABAJO
	TALLERES DESARROLLADOS EN EL COMPONENTE APE DESDE EL CAP 4 HASTA EL	5	TALLERES
C97	FINAL		
	APLICACIÓN EN SCALA UTILIZANDO PROGRAMACIÓN FUNCIONAL Y ORIENTADA A OBJETOS	5	EXAMENES
	EXAMEN FINAL	25	EXAMENES
C98	SUSPENSIÓN		
	Total:	100	

TEXTOS U OTRAS REFERENCIAS REQUERIDAS PARA EL APRENDIZAJE DE LA ASIGNATURA

Libros, revistas, bases digitales, periódicos, direcciones de Internet y demás fuentes de información, pertinentes y actuales.

BÁSICA

- | |
|--|
| 1. Tucker, A., Lenguajes de Programación Principios y Paradigmas, 1ra, 2011, McGraw Hill Interamericana S.L. |
| 2. Odersky, Martin, Lex Spoon, and Bill Venners. "Programming in Scala: a comprehensive stepby-step guide." Artima Inc, 2012 |

COMPLEMENTARIA

Esta asignatura no tiene bibliografía complementaria

Docente: MEJIA PESANTEZ PIEDAD MAGALI

Director: VEINTIMILLA REYES JAIME EDUARDO

Finalizado: 17/3/2024

Publicado: 27/3/2024