

SÍLABO



FACULTAD DE INGENIERÍA

MARZO 2024-AGOSTO 2024

NOMBRE DE LA ASIGNATURA		CÓDIGO:	17936
PROGRAMACIÓN I: ALGORITMOS, DATOS Y ESTRUCTURAS - GRUPO: 1			
CARRERA	COMPUTACION REDISEÑO		
CICLO O SEMESTRE	SEGUNDO NIVEL	EJE DE FORMACIÓN	BÁSICAS, 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):

ESTRELLA MALDONADO RENE GUSTAVO - (R.E.)	(rene.estrella@ucuenca.edu.ec)	PRINCIPAL
--	----------------------------------	-----------

DESCRIPCIÓN DE LA ASIGNATURA:

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

La Carrera de Computación trae consigo desafíos constantes dado su vertiginoso avance. Dentro del área de la Computación, la programación juega un rol importante, dada la necesidad de desarrollar aplicaciones informáticas para una gran variedad de dominios. Los profesionales en el área de la Informática, y específicamente, los Ingenieros en Computación son los llamados a satisfacer esta demanda de la sociedad actual. Es por ello que la malla curricular de la Carrera de Ingeniería en Computación incluye una serie de asignaturas que permiten a los estudiantes prepararse en el área de la programación de computadoras. El proceso de aprendizaje dentro de esta materia se centra esencialmente en dos áreas: (1) formulación de algoritmos, y (2) implementación de dichos algoritmos en un lenguaje de programación. Esta asignatura proporciona a los estudiantes de la carrera una formación en el formulación de algoritmos como fase previa al desarrollo de software, y los conocimientos de programación necesarios para implementar dichos algoritmos mediante el uso del lenguaje C.

REQUISITOS DE LA ASIGNATURA

Esta asignatura no tiene co-requisitos

PRE-REQUISITOS	
Asignatura	Código
CÁLCULO DIFERENCIAL	18581
ÁLGEBRA LINEAL	18580

OBJETIVO(S) DE LA ASIGNATURA:

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

Objetivo general: Aprender a analizar, diseñar, implementar y probar soluciones relacionadas con algoritmos que incluyan estructuras de datos para resolver un problema de un dominio dado utilizando un lenguaje de alto nivel como C.

Objetivos específicos:

1. Estudiar una metodología básica que permita al estudiante entender la resolución de un problema de manera ordenada, con pasos claramente diferenciados, que creen hábitos adecuados al momento de resolver un problema.
2. Entender el funcionamiento básico de la memoria del computador así como también del procesador, a fin de introducir al estudiante en conceptos primordiales para la definición de variables y estructuras de datos.
3. Aplicar la metodología estudiada para la formulación de algoritmos representados a través de pseudocódigo, de tal forma que el estudiante describa la solución que va a implementar desde un nivel alto.
4. Formular las soluciones de problemas en diversos ámbitos mediante algoritmos que puedan ser implementados en un lenguaje de programación.
5. Diseñar programas de computadora especificando claramente los recursos computacionales que van a ser utilizados (variables, estructuras, etc.)
6. Entender los conceptos de tipos de datos, estructuras de datos, librerías y funciones independientemente del lenguaje de programación utilizado.
7. Ser capaz de implementar algoritmos de complejidad básica e intermedia en el lenguaje de programación C.

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. Resuelve problemas mediante la aplicación de una metodología sistemática que permita diseñar soluciones en forma de algoritmos.	<ul style="list-style-type: none"> • El estudiante será capaz de resolver los ejercicios planteados aplicando las diferentes fases de la metodología estudiada. 	<ul style="list-style-type: none"> • Resolución de ejercicios sobre los temas estudiados en clase.
RdA2. Conoce el funcionamiento básico de una computadora en lo que se refiere al almacenamiento de datos en RAM, la ejecución de un programa en el CPU, y la presentación de resultados en periféricos de salida.	<ul style="list-style-type: none"> • El estudiante realizará adecuadamente las prácticas, resolviendo problemas y teniendo claras las consideraciones y limitaciones del hardware del computador. Cabe destacar que este conocimiento será acotado a lo necesario para entender la programación subsiguiente. 	<ul style="list-style-type: none"> • Resolución de ejercicios sobre los temas estudiados en clase.
RdA3. Aplica la metodología para solucionar un problema planteado mediante un algoritmo representado como diagramas de flujo o pseudocódigo, antes de implementarlo en un lenguaje de programación.	<ul style="list-style-type: none"> • El estudiante realizará de forma adecuada los ejercicios propuestos y las prácticas establecidas para que el resultado de aprendizaje evidencie su entendimiento e involucramiento con la materia. • el estudiante demostrará sus destrezas a la hora realizar tanto los ejercicios en clases como las prácticas y trabajo autónomo, relacionado con el análisis el problema, ya sea este numérico, de texto, estructuras, archivos. 	<ul style="list-style-type: none"> • Resolución de ejercicios mediante el planteamiento de algoritmos utilizando diagramas de flujo o pseudocódigo.
RdA4. Comprende los conceptos de tipos de datos, estructuras de datos, librería y función desde un punto de vista independiente al lenguaje de programación.	<ul style="list-style-type: none"> • Estos conceptos se verán reflejados en prácticas y trabajo autónomo a ser presentado al profesor de la asignatura. 	<ul style="list-style-type: none"> • Resolución de ejercicios mediante el planteamiento de algoritmos utilizando diagramas de flujo o pseudocódigo.
RdA5. Implementa los algoritmos planteados en un lenguaje de programación, adquiriendo destrezas en su uso y entendiendo que el análisis que se ha realizado de un problema del mundo real puede ser resuelto mediante el diseño e implementación de un algoritmo.	<ul style="list-style-type: none"> • El estudiante realizará las prácticas en clases, trabajo autónomo y pruebas necesarias de tal forma que se evidencie un entendimiento tanto de la formulación de algoritmos como de su implementación en el lenguaje de programación C. 	<ul style="list-style-type: none"> • Resolución de ejercicios mediante el diseño de algoritmos y su implementación en el lenguaje de programación C.

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 BÁSICOS			

SUB-UNIDADES	Nro. SESIÓN	COMPONENTE DE APRENDIZAJE	ACTIVIDADES DE APRENDIZAJE	
1. EVALUACIÓN DIAGNÓSTICA 2. Presentación del sílabo 3. INTRODUCCIÓN A LA PROGRAMACIÓN 4. CONCEPTOS BÁSICOS	1	APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	Presentación de la materia	1 horas
			Conceptos básicos	1 horas
		APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	Revisión sobre conceptos fundamentales en programación.	1 horas
		APRENDIZAJE AUTÓNOMO (AA)	Revisión sobre conceptos fundamentales en programación.	1 horas
2. METODOLOGÍA BÁSICA PARA LA RESOLUCIÓN DE PROBLEMAS				
1. RESOLUCIÓN DE PROBLEMAS MEDIANTE PROGRAMAS DE COMPUTADORA 2. METODOLOGÍA PARA LA RESOLUCIÓN DE PROBLEMAS	2	APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	Metodología para la resolución de problemas mediante programas de computadora	2 horas
		APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	Aplicación de la metodología para resolver problemas.	1 horas
		APRENDIZAJE AUTÓNOMO (AA)	Aplicación de la metodología para resolver problemas.	1 horas
3. ALGORITMOS				
1. CONCEPTO DE VARIABLE 2. CONCEPTO DE ALGORITMO 3. REPRESENTACIÓN DE UN ALGORITMO: DIAGRAMAS DE FLUJO, PSEUDOCÓDIGO 4. IMPORTANCIA DE DOCUMENTAR UN ALGORITMO 5. PRUEBA DE ESCRITORIO	3	APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	Definición de Algoritmo	1 horas
			Formas de representación de algoritmos	1 horas
			Diagramas de Flujo	1 horas
			Variables	1 horas
		APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	Planteamiento de algoritmos simples usando diagramas de flujo	2 horas
		APRENDIZAJE AUTÓNOMO (AA)	Planteamiento de algoritmos simples	10 horas
4. ESTRUCTURAS DE CONTROL DEL FLUJO DE EJECUCIÓN				

SUB-UNIDADES	Nro. SESIÓN	COMPONENTE DE APRENDIZAJE	ACTIVIDADES DE APRENDIZAJE	
1. ESTRUCTURA DE SELECCIÓN (CONDICIONAL) 2. ESTRUCTURA DE REPETICIÓN (BUCLE)	4	APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	Flujo de ejecución de un algoritmo/programa	1 horas
			Estructuras de Control: Selección	7 horas
			Estructuras de Control: Repetición	7 horas
			Pruebas de Escritorio	1 horas
			Representación de algoritmos usando pseudocódigo	8 horas
		APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	Planteamiento de algoritmos utilizando estructuras de selección	4 horas
			Planteamiento de algoritmos usando estructuras de selección y repetición	4 horas
			Representación de algoritmos usando pseudocódigo	4 horas
		APRENDIZAJE AUTÓNOMO (AA)	Planteamiento de algoritmos utilizando estructuras de selección	12 horas
			Planteamiento de algoritmos usando estructuras de selección y repetición	12 horas
			Representación de algoritmos usando pseudocódigo	12 horas
		5. IMPLEMENTACIÓN DE ALGORITMOS USANDO UN LENGUAJE DE PROGRAMACIÓN		
1. CARACTERÍSTICAS DEL LENGUAJE C 2. TIPOS DE DATOS SOPORTADOS EN C 3. CONSTANTES Y VARIABLES 4. DECLARACIONES Y ASIGNACIONES 5. ESTRUCTURAS DE CONTROL DEL FLUJO DE EJECUCIÓN	5	APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	Introducción al lenguaje de programación C	1 horas
			Variables y Tipos de Datos	1 horas
			Estructuras de Selección en el lenguaje C	1 horas
			Estructuras de Repetición en el lenguaje C	1 horas
		APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	Implementación de algoritmos en C	2 horas
		APRENDIZAJE AUTÓNOMO (AA)	Implementación de algoritmos en C	6 horas
6. MODULARIZACIÓN				
1. IMPORTANCIA DE LA MODULARIZACIÓN 2. FUNCIONES, PARÁMETROS Y PASO DE PARÁMETROS	6	APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	Resolución de problemas grandes y complejos en programación	2 horas
			Particionamiento de un problema simple en tareas manejables	2 horas
		APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	Diseño y programación de algoritmos que involucren el uso de funciones	2 horas
		APRENDIZAJE AUTÓNOMO (AA)	Diseño y programación de algoritmos que involucren el uso de funciones	6 horas
7. ESTRUCTURAS DE DATOS				

SUB-UNIDADES	Nro. SESIÓN	COMPONENTE DE APRENDIZAJE	ACTIVIDADES DE APRENDIZAJE	
1. ARREGLOS UNIDIMENSIONALES 2. ARREGLOS BIDIMENSIONALES	7	APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	Arreglos Unidimensionales	4 horas
			Cadenas de Texto	4 horas
			Arreglos Bidimensionales	4 horas
		APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	Implementación de algoritmos que involucren el uso de arreglos unidimensionales	2 horas
			Implementación de algoritmos que involucren cadenas de texto	2 horas
			Implementación de algoritmos que involucren arreglos bidimensionales	2 horas
		APRENDIZAJE AUTÓNOMO (AA)	Implementación de algoritmos que involucren el uso de arreglos unidimensionales	6 horas
			Implementación de algoritmos que involucren cadenas de texto	6 horas
			Implementación de algoritmos que involucren arreglos bidimensionales	6 horas
8. BÚSQUEDAS Y ORDENAMIENTOS				
1. BUSQUEDA SECUENCIAL Y BINARIA 2. FORMAS DE ORDENAMIENTO BÁSICAS: BURBUJA, POR INSERCIÓN Y POR SELECCIÓN 3. EJERCICIOS EN CLASE 4. PRÁCTICAS 5. TRABAJO AUTÓNOMO	8	APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	Algoritmos de Búsqueda: Búsqueda Lineal	2 horas
			Algoritmos de Búsqueda: Búsqueda Binaria	2 horas
			Algoritmos de Ordenamiento: Selección	1 horas
			Algoritmos de Ordenamiento: Inserción	1.5 horas
			Algoritmos de Ordenamiento: Burbuja	1.5 horas
		APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	Programación de algoritmos utilizando búsqueda lineal y binaria	2 horas
			Implementación de algoritmos de ordenamiento	2 horas
		APRENDIZAJE AUTÓNOMO (AA)	Programación de algoritmos utilizando búsqueda lineal y binaria	6 horas
			Implementación de algoritmos de ordenamiento	6 horas
9. GESTIÓN DE ARCHIVOS				
1. ARCHIVOS Y SUS TIPOS 2. CREACIÓN DE ARCHIVOS DE TEXTO 3. LECTURA DE ARCHIVOS, MODOS DE LECTURA 4. MODIFICACIÓN DE ARCHIVOS DE TEXTO 5. ELIMINACIÓN DE ARCHIVOS	9	APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	Acceso a archivos de texto	2 horas
			Lectura y escritura de archivos de texto	2 horas
		APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	Implementación de algoritmos que involucren la gestión de archivos de texto	2 horas
		APRENDIZAJE AUTÓNOMO (AA)	Implementación de algoritmos que involucren la gestión de archivos de texto	6 horas
		APRENDIZAJE EN CONTACTO CON EL DOCENTE (ACD)	64 horas	
		APRENDIZAJE PRÁCTICO EXPERIMENTAL - ASIGNATURA (APE/A)	32 horas	
		APRENDIZAJE AUTÓNOMO (AA)	96 horas	

RECURSOS O MEDIOS PARA EL APRENDIZAJE

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

- Aula virtual como repositorio de presentaciones, ejemplos desarrollados, envío y recepción de trabajos y evaluaciones en línea.
- Bibliografía, documentos y manuales disponibles en Internet.
- Entorno de desarrollo que soporte el lenguaje de programación C.

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
EXAMENES	50
TAREAS EN CLASE	10
PRUEBAS	40
TOTAL:	100

	DETALLE DE CRITERIOS DE ACREDITACIÓN	PUNTAJE / CRITERIO GENERAL	
C94	APROVECHAMIENTO I		
	Resolución de ejercicios en clase y fuera de clase mediante el planteamiento de algoritmos.	5	TAREAS EN CLASE
	Prueba sobre el planteamiento de algoritmos usando diagramas de flujo que involucren estructuras de selección y repetición	10	PRUEBAS
	Prueba sobre el planteamiento de algoritmos usando pseudocódigo que involucren estructuras de selección y repetición	10	PRUEBAS
C95	INTERCICLO		
	Examen interciclo	20	EXAMENES
C96	APROVECHAMIENTO II		
	Prueba sobre la implementación de algoritmos en lenguaje C que involucren arreglos unidimensionales, funciones, búsquedas y ordenamientos	10	PRUEBAS
	Resolución de ejercicios en clase y fuera de clase mediante el planteamiento de algoritmos.	5	TAREAS EN CLASE
	Prueba sobre la implementación de algoritmos en lenguaje C que cadenas de texto, arreglos bidimensionales y archivos	10	PRUEBAS
C97	FINAL		
	Examen final	30	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. Deitel, P. y Deitel, H. C How to Program. Pearson Education.

COMPLEMENTARIA

Esta asignatura no tiene bibliografía complementaria

Docente: ESTRELLA MALDONADO RENE GUSTAVO

Finalizado: 20/3/2024

Director: VEINTIMILLA REYES JAIME EDUARDO

Publicado: 27/3/2024