



**FACULTAD DE INGENIERIA Y C. AGROPECUARIA**  
**INGENIERIA EN SISTEMAS DE COMPUTACIÓN E INFORMATICA**  
**ACI220 (PROGRAMACION ESTRUCTURADA)**  
**2015 - 2**  
**MARZO 2015 - JULIO 2015**

**1. Identificación.- (sílabo maestro)**

Número de sesiones: 64  
Número total de hora de aprendizaje: 147 h  
Créditos – malla actual: 4  
Profesor: Anita Yáñez  
  
Correo electrónico del docente (Udlanet): a.yanez@udlanet.ec  
Coordinador: Ing. Marco Galarza  
Campus: Queri  
Pre-requisito: Co-requisito:  
Paralelo: 1

Tipo de asignatura:

Optativa	
Obligatoria	X
Práctica	

Organización curricular:

Unidad 1: Formación Básica	
Unidad 2: Formación Profesional	X
Unidad 3: Titulación	

Campo de formación:

Campo				
Fundamentos teóricos	Praxis profesional	Epistemología y metodología de la investigación	Integración de saberes, contextos y cultura	Comunicación y lenguajes
	X			

**2. Descripción del curso.- (sílabo maestro)**

El presente curso abarca el desarrollo lógico de un algoritmo desde la etapa inicial de análisis utilizando representación gráfica (Diagramas de Flujo) y pseudocódigo además del uso de métodos formales de verificación de programas (pruebas de escritorio), hasta su implementación en un lenguaje de programación estructurado (lenguaje C), utilizando estructuras de control, funciones, arreglos, estructuras, programación modular y archivos.

**3. Objetivo del curso.- (sílabo maestro)**

Proporcionar conocimiento teórico y práctico para desarrollar soluciones lógicas a problemas computacionales básicos, utilizando conceptos, métodos y técnicas estructuradas. Desde el diseño del algoritmo al programa ejecutable en un lenguaje de programación estructurado.

**4. Resultados de aprendizaje deseados al finalizar el curso: (sílabo maestro)**

Resultados de aprendizaje (RdA)	RdA perfil de egreso de carrera	Nivel de dominio (carrera)
1. Desarrolla pensamiento lógico para estructurar la secuencia de pasos requeridos para una solución algorítmica.	Aplica el enfoque sistémico en el análisis y resolución de problemas relacionados con las TIC.	I <u>X</u> M ____ F ____
2. Implementa algoritmos computacionales modulares en un lenguaje de programación estructurado.	Aplica metodologías de investigación, pensamiento lógico, fundamentos matemáticos, principios algorítmicos y teorías de Ciencias de la Computación en la fundamentación, modelación y diseño de soluciones informáticas.	I <u>X</u> M ____ F ____
3. Aplicar el lenguaje estructurado para la manipulación de archivos con el objeto de almacenamiento y manejo de información.		

## 5. Sistema de evaluación.- (sílabo maestro)

De acuerdo al Modelo Educativo de la UDLA la evaluación busca evidenciar el logro de los resultados de aprendizaje (RdA) enunciados en cada carrera y asignatura, a través de mecanismos de evaluación (MdE). Por lo tanto la evaluación debe ser continua, formativa y sumativa.

La evaluación es continua, formativa y sumativa.

Reporte de progreso 1: 35%  
 Reporte de progreso 2: 35%  
 Evaluación final: 30%

Al finalizar el curso habrá un examen de recuperación para los estudiantes que deseen reemplazar la nota de un examen anterior (ningún otro tipo de evaluación). Este examen es de carácter complejo y de alta exigencia, por lo que el estudiante necesita prepararse con rigurosidad. La nota de este examen reemplazará a la del examen que sustituye. Para rendir el Examen de Recuperación, es requisito que el estudiante haya asistido por lo menos al 80% del total de las sesiones programadas de la materia.

## 6. Metodología del curso y de mecanismos de evaluación.- Docente

Las metodologías que se aplicarán para el desarrollo del curso son:

1. Lectura Previa: Para cada unidad el docente facilitará fuentes de información que el estudiante deberá leer previa la clase.
2. Prácticas individuales en laboratorio: Explicado el tema el docente, propone a los alumnos una actividad de aplicación de dichos contenidos.
3. Clase magistral: El docente presentará el tema a tratar de manera resumida, usando medios tecnológicos, (diapositivas), se usará la técnica pregunta – respuesta.
4. Trabajos Individuales: Que serán desarrollados dentro y fuera del aula para reforzar lo aprendido en clase.
5. Trabajo colaborativo: Los estudiantes realizarán trabajos en grupo dentro y fuera del aula de clases.
6. Debate en Clase: Los estudiantes podrán dar sus opiniones en clase de asuntos pertinentes al curso.

Componentes de la libreta de calificaciones

Progreso 1: 35%



El progreso 1 consta de las siguientes tareas con su porcentaje de evaluación:

- a) Portafolio De Prácticas de laboratorio 20%, de acuerdo a las guías de laboratorio y rúbricas correspondientes.
- b) Portafolio de Ejercicios 10%, se adjuntará rúbrica.
- c) Prueba teórica 30%.
- d) Examen- Proyecto Práctico Progreso 1, 40% se adjuntará rúbrica

*Progreso 2: 35%*

El progreso 2 consta de las siguientes tareas con su porcentaje de evaluación:

- a) Portafolio De Prácticas de laboratorio 10%, de acuerdo a las guías de laboratorio y rúbricas correspondientes.
- b) Portafolio de Ejercicios 20%. se adjuntará rúbrica
- c) Examen, teórico 30%, se adjuntará rúbrica
- d) Examen- Proyecto Práctico Progreso 2, 40% se adjuntará rúbrica

*Evaluación Final 30%*

La evaluación final consta de las siguientes tareas con su porcentaje de evaluación.

- a) Examen- Proyecto Práctico Evaluación Final 60%, se adjuntará rúbrica
- b) Proyecto Final, 40% se adjuntará rúbrica

Cada ítem de evaluación contendrá la rúbrica respectiva, publicada en el apoyo virtual junto con tarea enviada.

## 7. Temas y subtemas del curso.- (sílabo maestro)

RDA	TEMA	SUB TEMA
1. Desarrolla pensamiento lógico para estructurar la secuencia de pasos requeridos para una solución algorítmica.	1. Algoritmia	1.1. Evolución de los paradigmas de programación
		1.2 Lenguajes de programación - Programación estructurada
		1.3. Proceso de programación
		1.4. Algoritmos
		1.5. Elementos de la programación. (evaluación de expresiones, tipos de datos, operadores)
		1.6.Diseño de algoritmos
		1.7 Representación gráfica DF
		1.8 Pseudocódigo
		1.9 Métodos formales para verificación de programas
		1.10 Sentencias secuenciales
		1.11 Sentencias condicionales y selectivas
		1.12 Sentencias repetitivas
2. Implementa algoritmos computacionales modulares en un lenguaje de programación estructurado.	2. Estructuras de control del lenguaje de programación C.	2.1. Elementos del lenguaje de programación en C
		2.2. Sentencias de control simples

		2.3 Sentencias de control condicionales y selectivas
		2.4 Sentencias de control Iterativas
		2.5 Arreglos, Unidimensionales - bidimensionales. Cadenas
		2.6 Estructuras
	3. Programación modular	3.1 Funciones
		3.2. Punteros
3. Aplica el lenguaje estructurado para la manipulación de archivos con el objeto de almacenamiento y manejo de información.	4. Entrada / Salida de información	4.1 Manejo de archivos

## 8. Planificación secuencial del curso.- Docente

SEMANA 1 a la 4 (9-03 al 03-03)					
RdA	Tema	Sub tema	Actividad/ metodología/clase	Tarea/autónoma	MdE/Producto/
1	1. Algoritmia	1.1. Evolución de los paradigmas de programación	Instrucción directa	Lectura del libro Introducción a la programación Lógica y Diseño( Farrell, 2013, Capítulo 1, 2, 3, 4)	Prueba Teórico control de lectura Fecha de entrega: 2 de Abril.
		1.2 Lenguajes de programación – Programación estructurada	Portafolio de Ejercicios	Portafolio de ejercicios. Introducción a la programación Lógica y Diseño( Farrell, 2013, Capítulo 1, 2, 3, 4)	Portafolio de Ejercicios: Fecha de entrega: 2 de Abril.
		1.3. Proceso de programación	Practicas Individuales en el laboratorio	Portafolio de prácticas	Portafolio De Prácticas de laboratorio Fecha de entrega: 25 de Marzo
		1.4. Algoritmos			
		1.5. Elementos de la programación. (evaluación de expresiones, tipos de datos, operadores)			
		1.6.Diseño de algoritmos			
		1.7 Representación gráfica DF			
		1.8 Pseudocódigo			
		1.9 Métodos formales para verificación de programas			
		1.10 Sentencias secuenciales			

		1.11 Sentencias condicionales y selectivas			
		1.12 Sentencias repetitivas			
SEMANA 5 - 7 (06-04 al 17-04)					
2	2. Estructuras de control del lenguaje de programación C.	2.1. Elementos del lenguaje de programación	Instrucción directa		
		2.2. Sentencias lineales	Portafolio de Ejercicios	Portafolio de ejercicios. Una Introducción a la Programación Estructurada en C. Ruiz, R, 2013).	Portafolio de Ejercicios: Fecha de entrega: 16 de Abril.
EVALUACIÓN DEL PROGRESO 1 : 20 al 24 de Abril					
SEMANA 8 a la 13 (27-04 al 29-05)					
2	2. Estructuras de control del lenguaje de programación C.	2.3 Sentencias de control condicionales y selectivas	Instrucción directa	Lectura del libro: Una Introducción a la Programación Estructurada en C. Ruiz, R, 2013. Capítulo 1,2, y 3).	Prueba Teórico control de lectura Fecha de entrega: 26 de Mayo
		2.4 Sentencias de control Iterativas		Lectura; Una Introducción a la Programación Estructurada en C. Ruiz, R, 2013. Capítulo 4, y 6). Portafolio de prácticas	
		2.5 Arreglos, Unidimensionales - bidimensionales. Y cadenas	Portafolio de prácticas	Prácticas Individuales en el laboratorio	Portafolio de Prácticas de laboratorio Fecha de entrega: 19 de Mayo
		2.6 Estructuras	Portafolio de Ejercicios	Portafolio de ejercicios. Una Introducción a la Programación Estructurada en C. Ruiz, R, 2013).	Portafolio de Ejercicios: Fecha de entrega: 29 de Mayo
	3. Programación modular	3.1. Funciones			Examen Práctico Progreso 2 Del 1 al 25 de Junio



**EVALUACIÓN DEL PROGRESO 2 : 1 al 5 de Junio**

**SEMANA 14 a la 16 (8 al 16 de Junio)**

		3.2 Punteros			
3	4. Entrada / Salida de información	4.1 Manejo de archivos	Instrucción directa	Examen Final	Examen Práctico Evaluación Final Fecha : 7 de Julio
				Proyecto Final Integrador	Proyecto Final Integrador de la materia Fecha : 14 de Julio

**EVALUACIÓN DEL FINAL del 6 al 10 de Julio**

**PRESENTACIÓN PROYECTO FINAL DEL 13 al 17 de Julio**

#### 9. Observaciones generales.- Docente

1. Las tareas serán evaluadas en el aula virtual y del dentro del plazo establecido, por ningún motivo se recibirán fuera del apoyo virtual.
2. Se tomaré lista dentro de los primeros 5 minutos luego de iniciado cada módulo, si el estudiante llega después, podrá ingresar de forma silenciosa, pero no se registrará la asistencia.
3. La copia de ejercicios, exámenes, proyectos, y todas las actividades de aprendizaje solicitadas por el docente, serán sancionados de acuerdo al reglamento de la universidad, y se calificará con la mínima calificación (cero).
4. Se restringe durante el desarrollo de clases el uso de celulares, redes sociales y audífonos.
5. Los alimentos dentro del aula no están permitidos.
6. El estudiante puede acceder a tutoría personal en los horarios establecidos por el docente.
7. En el caso de inasistencia es responsabilidad del estudiante igualarse en los contenidos de la materia dictada en dicha clase.
8. En el caso de que un estudiante falte a una sesión en la que se realicen pruebas o prácticas de laboratorio, no se podrán recuperar las calificaciones. Las fechas de las evaluaciones serán publicadas en el apoyo virtual de la materia.

#### 10. Referencias bibliográficas.- (sílabo maestro)

- Farrell, J. (2013). *Introducción a la programación Lógica y Diseño*. ( 7a. ed ) México, Mexico: Cengage Learning Editores. ISBN 978-1-133-52651-3.
- Ruiz, R. (2013). *Una Introducción a la Programación Estructurada en C*. ( 1ra ed ) Argentina: El Cid Editor.

##### 10.1. Referencias complementarias.- Docente

- Juganaru, M. (2014). *Introducción a la Programación*. (1ra ed) Mexico, Mexico. Larousse Grupo Editorial Patria. ISBN ebook: 978-607-438-920-3.
- Márquez, G., Osorio, S., & Olvera, N. (2010). *Introducción a la Programación Estructurada en C*. (1ra ed) México, México: Pearson Educación. ISBN 978-607-32-0600-6