



FACULTAD DE INGENIERÍA Y CIENCIAS AGROPECUARIAS
INGENIERÍA EN SISTEMAS DE COMPUTACIÓN E INFORMÁTICA
ACI 320 PROGRAMACIÓN ORIENTADA A OBJETOS
Período: 2014 - 2
Marzo – Junio 2014

1. Identificación

Número de Sesiones 646
Número de horas_80_ o Créditos_6__
Profesor(a): Xavier Armendariz
Correo electrónico del docente: x.armendariz@udlanet.ec
Coordinador (a): Marco Galarza
Campus: Sede Queri
Pre-requisito ACI-220
Co-requisito
Paralelo:

2. Descripción de la Asignatura:

Este curso introduce al estudiante a la solución de problemas por medio del diseño y la programación orientada a objetos. El énfasis está en el análisis del problema y el diseño de la solución, la documentación y la implementación. Los alumnos utilizan bibliotecas para crear proyectos de software, utilizando java como lenguaje de programación para realizar los ejercicios propuestos en clase

3. Objetivo de la Asignatura:

Programar aplicaciones de software utilizando técnicas orientadas a objetos a través del diseño de clases extensibles y robustas expresadas en diseños UML e implementadas utilizando interfaces cooperativas.

4. Resultados de Aprendizaje (RdA) deseados:

Al completar este curso, se espera que el estudiante::

1. Implemente programas orientados a objetos empleando JAVA.
2. Diseñe clases utilizando técnicas orientadas a objetos usando el lenguaje UML.
3. Implemente programas que muestren un comportamiento especificado utilizando el modelo orientado a objetos y optimizando el código de programación.

4. Programe con creatividad aplicaciones orientadas a objetos utilizando interfaces de entrada y salida a través de un lenguaje de programación.

5. Sistema de evaluación

En términos generales, la Universidad de Las Américas estipula la siguiente distribución porcentual para las evaluaciones previstas en cada semestre. Recordar que las Cátedras se pueden evaluar a través de proyectos y que la herramienta de evaluación debe ser la rúbrica, y que los Controles deben ser ejercicios y tareas diversas a lo largo del semestre.

Cátedra 1: 20%
 Cátedra 2: 25%
 Controles: 25%
 Examen Final: 30%

6. Unidades didácticas/ Desarrollo secuencial del curso

Resultados de Aprendizaje	N./Unidad Tema	Subtemas
Diseñe clases utilizando técnicas orientadas a objetos usando el lenguaje UML.	1. Diseño de clases	1.1 Características de la POO 1.2 Clases y objetos 1.3 Diagramas UML de Clases 1.4 Relaciones entre Clases 1.5 Estructuras Comunes de Clases 1.6 Modelado de Clases en UML utilizando un editor
Implemente programas orientados a objetos empleando JAVA.	2. Aplicaciones de JAVA	2.1 Entorno de programación JAVA usando un IDE 2.2 Convenciones de escritura de código 2.3 Manejo de excepciones 2.4 Depuración de programas
Implemente programas que muestren un comportamiento especificado utilizando el modelo orientado a objetos y optimizando el código de programación.	3. Implementación de clases 4. Colecciones 5. Diseño avanzado de clases	3.1 Definiendo Clases 3.2 Implementación de asociaciones entre clases 3.3 Herencia 3.4 Prueba de Unidades 4.1 Arreglos estáticos 4.2 Arreglos dinámicos genéricos(ArrayList <E>) 5.1 Clases Abstractas 5.2 Polimorfismo 5.3 Interfaces
Programe con creatividad aplicaciones orientadas a objetos utilizando interfaces de entrada y salida a través de	6. Implementación Avanzada de Clases	6.1 Programación de E/S 6.2 Interfaz gráfica de usuario

un lenguaje de programación		
-----------------------------	--	--

6.1 Unidades didácticas/ Evidencias de aprendizaje

N. / Unidad	Sesiones	Estrategias metodológicas/ Actividades	Tareas/ Trabajo autónomo	Mecanismos de evaluación/ evidencias de aprendizaje	Ponderación % - puntos
1. Diseño de clases	16	1.1 – 1.5 Clases magistrales 1.6 Taller de implementación usando eclipse	1.1 -1.5 Lecturas: (Barker, 2006), (Flores, 2012) Laboratorios de trabajos prácticos	1.1 -1.6 Taller- Producir diagramas de clase UML que modelen asociaciones entre clases Producir un diagrama de clase UML a partir de una especificación, que muestre: Clases, Atributos, Métodos, Relaciones.	5%
2. Aplicaciones de JAVA	10	2.1-2.4 Taller de implementación de clases, documentación y excepciones	2.1-2.4 Lecturas: (Barker, 2006), (Flores, 2012) Laboratorios de trabajos prácticos	2.1-2.4 Producir una aplicación basada en menús que lea datos del teclado y despliegue resultados en la salida estándar Producir una aplicación que utilice: -Clase de envoltura para leer datos numéricos -Excepciones para manejar datos mal formados -Estructuras de control para controlar la lectura de los datos	5%
CATEDRA I					20%
3. Implementación de clases	12	3.1 - 3.4 Clases magistrales Taller de implementación de herencia y test de unidades	3.1 - 3.4 Lecturas: (Barker, 2006), (Flores, 2012) Laboratorios de trabajos prácticos	3.1-3.4 Producir una clase de aplicación de prueba que verifique la implementación de una clase de Java Implementación de constructores, selectores y modificadores de una clase de Java	2.5%

				Uso de herencia para implementar una relación de especialización/generalización	
4. Colecciones	8	4.1 – 4.2 Clases magistrales Taller de implementación de colecciones	4.1 – 4.2 Laboratorios de trabajos prácticos	4.1 – 4.2 Creación de una colección: A partir de un conjunto de objetos, de un arreglo, Procesamiento de colecciones: -Encontrar un elemento de la colección que cumple con una característica específica. -Contar el número de elementos de una colección que cumplen con una característica específica. -Analizar los contenidos de una colección -Eliminar elementos de una colección	1.5%
5. Diseño avanzado de clases	8	5.1 – 5.3 Clases magistrales Taller de implementación de clases abstractas, polimorfismo e interfaces	5.1 – 5.3 Lecturas: (Barker, 2006), (Flores, 2012) Laboratorios de trabajos prácticos	5.1 – 5.3 Programas orientado a objetos que contenga: Clases abstractas, polimorfismo, interfaces, clases que implementen interfaces y clases abstractas	1%
CATEDRA II					25%
6. Implementación Avanzada de Clases	8	6.1 - 6.2 Taller de implementación de aplicaciones de E/S	6.1 - 6.2 Laboratorios de trabajos prácticos	6.1 - 6.2 Producir aplicaciones que lean datos de un archivo y los analicen Producir aplicaciones que escriban datos en un archivo	5%
EXAMEN FINAL					35%
CONFRONTACIÓN					

7. Metodología para el desarrollo de la asignatura

De acuerdo con la naturaleza del curso, sus contenidos serán desarrollados en diferentes niveles de aprendizajes desde la adquisición de conocimientos básicos, su aplicación, análisis, síntesis y evaluación a través de actividades diseñadas para mejorar su aprendizaje. Se ejecutarán diferentes modalidades tales como:.

Clase magistral.

Estudios dirigidos.

Trabajo en grupos.

Prácticas en Laboratorio

8. Referencias

Héctor, F. (2012). *Programación Orientada a Objetos usando JAVA*. Bogotá: Ecoe Ediciones.

Jacqkie, B. (2006). *Beginning Java Objects*. New York: Apress.

9. Observaciones generales

- El estudiante debe preparar las temáticas planificadas previas a la ejecución de los talleres.
- Las clases se ejecutarán en el laboratorio de computación

Fecha: 06-03-2014

Elaborado por: Xavier Armendariz ____

Revisado por: Diego Buenaño ____

Aprobado por: Marco Galarza ____