



FACULTAD DE CIENCIAS DE LA ADMINISTRACIÓN

ESCUELA DE INGENIERÍA EN CIENCIAS DE LA COMPUTACIÓN

1. Datos generales

Materia: PROGRAMACIÓN II
Código: ICC305
Paralelo: A
Periodo: Septiembre-2023 a Febrero-2024
Profesor: SALGADO ARTEAGA JUAN CARLOS
Correo electrónico: jsalgado@uazuay.edu.ec

Nivel: 3

Distribución de horas.

Docencia	Práctico	Autónomo: 120		Total horas
		Sistemas de tutorías	Autónomo	
64	16	16	104	200

Prerrequisitos:

Código: ICC203 Materia: PROGRAMACIÓN I

2. Descripción y objetivos de la materia

Se busca que el estudiante desarrolle aplicaciones de uso general, empleando los fundamentos de la programación orientada a objetos, las bibliotecas de clases que forman parte de la Interfaz de Programación de Aplicaciones y un ambiente de desarrollo integrado.

Esta materia es un recurso de apoyo para toda la carrera; se relaciona con Programación III y con las materias que requieran el desarrollo de aplicaciones informáticas, como Bases de Datos, Teoría de Autómatas y Teoría de la Computación.

Programación II es una materia de carácter teórico - práctica. Constituye un curso básico en la formación de profesionales de la carrera de Ingeniería en Ciencias de la Computación ya que permite al alumno dominar las fases de resolución de problemas por computadora, además de capacitarlo en las técnicas de programación orientada a objetos. La asignatura se alinea con el ODS 4: "Educación de calidad", puesto que se plantea: Aplicar metodologías activas que fomenten el aprendizaje analítico, crítico y autónomo. Se vincula además con el ODS 9: "Industria, innovación e infraestructura", en cuanto se orienta a: Fomentar la generación de proyectos que permitan desarrollar aplicaciones de software destinadas a satisfacer necesidades sociales o empresariales.

3. Objetivos de Desarrollo Sostenible



4. Contenidos

01.01.	Paradigmas de programación
01.02.	Objetos. Clases Mensajes y Métodos
01.03.	Componentes del Modelo Objeto: Abstracción, Encapsulamiento, Modularidad. Jerarquía y Herencia.
01.04.	Relaciones cliente/servidor y de agregación. Polimorfismo.
01.05.	Ventajas de la Orientación a Objetos.
01.06.	La Programación Orientada a Objetos frente a la Programación Tradicional.
02.01.	Introducción. Historia. Definición de UML

02.02.	¿Qué forma un modelo de UML?
02.03.	Diagramas de UML Diagramas de clases.
02.04.	Elementos del Diagrama de Clases.
02.05.	Relaciones entre clases: Asociación, Composición, Agregación, Dependencia, Herencia, Ejemplos de relaciones entre clases.
03.01.	Clases: Declaración de Atributos, Implementación de Métodos.
03.02.	Objetos: Declaración, Instanciación, Inicialización de Objetos.
03.03.	Definición de mensaje. Partes que forman un mensaje.
03.04.	Operaciones de modificación, selección, iteración.
03.05.	Operaciones de construcción y destrucción
04.01.	Clases Públicas, Protegidas y Privadas.
04.02.	Derivación de Clases.
04.03.	Ámbito de clases bajo la derivación.
04.04.	Inicialización
04.05.	Asignación y Jerarquía.
04.06.	Herencia de Funciones.
05.01.	Invocación de los métodos de superclases desde objetos de subclases.
05.02.	Uso de referencias a superclases con variables de tipo subclase.
05.03.	Llamadas a métodos de subclases mediante variables tipo superclase
05.04.	Clases y métodos abstractos.
05.05.	Clases anidadas
06.01.	Jerarquía de datos
06.02.	Clase File
06.03.	Manipulación de archivos secuenciales
06.04.	Manipulación de archivos de acceso aleatorio.
07.01.	Introducción a bases de datos relacionales. Manejo de SQL.

5. Sistema de Evaluación

Resultado de aprendizaje de la carrera relacionados con la materia

Resultado de aprendizaje de la materia

Evidencias

ao1. Modela y diseña sistemas computacionales de diferente tamaño y complejidad con niveles de abstracción de acuerdo al contexto del problema, demostrando dominio del cuerpo de conocimiento.

-Aplica los conceptos fundamentales de la programación orientada a objetos. -Evaluación escrita
-Proyectos

-Diseña soluciones de acuerdo al paradigma orientado a objetos. -Evaluación escrita
-Proyectos

Desglose de evaluación

Evidencia	Descripción	Contenidos sílabo a evaluar	Aporte	Calificación	Semana
Evaluación escrita	Prueba escrita	Introducción a la Programación Orientada a Objetos , Lenguaje Unificado de Modelado	APORTE	6	Semana: 5 (16-OCT-23 al 21-OCT-23)
Proyectos	Resolución de ejercicios utilizando componentes del modelo objeto, diagramas en UML.	Introducción a la Programación Orientada a Objetos , Lenguaje Unificado de Modelado	APORTE	4	Semana: 5 (16-OCT-23 al 21-OCT-23)
Evaluación escrita	Prueba escrita	Herencia y derivación de clases. , Lenguaje de Programación Orientado a Objetos , Polimorfismo	APORTE	6	Semana: 10 (20-NOV-23 al 25-NOV-23)
Proyectos	Implementación de programación orientada a objetos utilizando estructuras de programación, herencia, derivación de clases y polimorfismo	Herencia y derivación de clases. , Lenguaje de Programación Orientado a Objetos , Polimorfismo	APORTE	4	Semana: 10 (20-NOV-23 al 25-NOV-23)
Evaluación escrita	Prueba escrita	Acceso a base de datos , Archivos y Flujos	APORTE	6	Semana: 15 (al)
Proyectos	Implementación de programación orientada a objetos utilizando archivos y bases de datos.	Acceso a base de datos , Archivos y Flujos	APORTE	4	Semana: 15 (al)
Evaluación escrita	Prueba escrita	Acceso a base de datos , Archivos y Flujos , Herencia y derivación de clases. , Introducción a la Programación Orientada a Objetos , Lenguaje Unificado de Modelado , Lenguaje de Programación Orientado a Objetos , Polimorfismo	EXAMEN	20	Semana: 19-20 (21-01-2024 al 27-01-2024)
Evaluación escrita	Prueba escrita	Acceso a base de datos , Archivos y Flujos , Herencia y derivación de clases. , Introducción a la Programación Orientada a Objetos , Lenguaje Unificado de Modelado , Lenguaje de Programación Orientado a Objetos , Polimorfismo	SUPLETORIO	20	Semana: 20 (al)

Metodología

Descripción	Tipo horas
Sobre cada tema significativo estudiado se proponen ejercicios que los estudiantes deben realizar. Estos ejercicios sirven para aplicar los conceptos impartidos o investigados de manera autónoma	Autónomo
El profesor realizará la exposición y explicaciones utilizando material de apoyo. Se propiciará la participación activa de los alumnos, se abrirá el foro para clarificación, y para motivar la participación de los estudiantes con sus opiniones e impresiones sobre los temas tratados.	Total docencia

Criterios de evaluación

Descripción	Tipo horas
Los ejercicios, trabajos prácticos e investigaciones deberán ser entregados en la fecha indicada para cada uno de ellos, se evaluará la aplicación del formato de presentación, redacción y ortografía, el desarrollo de las actividades propuestas, la participación y responsabilidad frente al trabajo en grupo, la honestidad intelectual. Las copias y/o plagios se sancionarán calificándolas con cero.	Autónomo
En las pruebas se evaluará el desarrollo de cada uno de los ítems propuestos, el dominio de los conocimientos y habilidades que se pretenden desarrollar	Total docencia

6. Referencias

Bibliografía base

Libros

Autor	Editorial	Título	Año	ISBN
HARVEY DEITEL Y PAUL DEITEL	Pearson Prentice Hall	JAVA HOW TO PROGRAM	2012	978-0-13-257566
GRADY BOOCH, JAMES RUMBAUGH, IVAR JACOBSON, JOSÉ SAEZ TRAD. JESÚS GARCIA REV. TEC.	Addison Wesley	LENGUAJE UNIFICADO DE MODELADO	2003	978-84-7829-028-4
BOOCH,G., ROBERT A. MAKSIMCHUK, MICHAEL W. ENGLE	Pearson Education	OBJECT-ORIENTED ANALYSIS AND DESIGN WITH APPLICATIONS	2007	0-201-89551-X
ROBER MARTIN	Pearson	UML PARA PROGRAMADORES JAVA	2004	9788420541099

Web

Software

Revista

Bibliografía de apoyo

Libros

Web

Autor	Título	Url
Oracle Technology Network	Java™ Platform, Standard Edition 8 API Specification	http://docs.oracle.com/javase/8/docs/api/index.html
Oracle Technology Network	JAVA SE Especifications	http://docs.oracle.com/javase/specs/
Oracle Technology Network	The Java Tutorials	http://docs.oracle.com/javase/tutorial/

Software

Autor	Título	Url	Versión
Oracle	Java SE	https://www.oracle.com/java/technologies/downloads/#java8	8u381
Oracle	NetBeans	http://web.archive.org/web/20180418102733/https://netbeans.org/downloads/index.html	8.2

Revista

Docente

Director/Junta

Fecha aprobación: **07/09/2023**

Estado: **Aprobado**