

ISW-311 – Programación II

Créditos: 6. Horas presenciales por semana: 8 (4 horas teoría, 4 horas prácticas)

Tiempo de estudio independiente por semana: 11.18 horas

Requisitos: ISW-211 Programación I

Descripción del Curso

Este curso está orientado a estudiantes del III nivel de Carrera. Es el segundo curso de programación donde se pretende que el estudiante vaya desarrollando su habilidad para resolver problemas de forma metódica y que cumpla con estándares de calidad.

Durante el curso el estudiante utilizará las técnicas de desarrollo de programas, aprendiendo a identificar el problema, analizarlo, y proceder a la descomposición del mismo para crear un modelo y diseñar la especificación del algoritmo apropiado que luego codificará utilizando el paradigma de la programación orientada a objetos con el lenguaje apropiado en el momento en que se imparte.

Objetivo general

- Resolver problemas de lógica con el apoyo del equipo de cómputo mediante el modelaje, diseño detallado y programación, aplicando técnicas actuales de desarrollo de software orientado a objetos y considerando criterios de calidad apropiados, con la finalidad de que se conozca la herramienta UML.

Objetivos específicos

1. Aplicar los conceptos básicos de programación orientada a objetos, mediante el desarrollo de clases de laboratorio, con la finalidad de que domine la herramienta y resuelva problemas específicos.
2. Aplicar las herramientas propias del lenguaje, para modelar las relaciones entre los objetos involucrados en la solución del problema, mediante el aprovechamiento de los laboratorios, para que se desenvuelva en un marco de programación y desarrollo de sistemas, acorde con las necesidades del medio.
3. Comprender el uso de interfaces gráficas de usuario, con el fin de comunicarse con ellos por medios amigables, mediante la utilización de las herramientas apropiadas en el espacio de laboratorio.
4. Comprender el entorno de la programación orientada a objetos, tomando en consideración el uso de datos abstractos, mediante lecturas y prácticas apropiadas, que permitan dominar los conceptos de referencia.
5. Manipular y aprovechar al máximo el trabajo con memoria dinámica, mediante la conceptualización adecuada para manipular eficientemente la información.

Contenidos

Unidad I. Relaciones entre objetos.

- Relación de Asociación simple de objetos.
- Relación de Agregación de objetos.
- Relación de Composición de objetos.

Unidad II. Herencia y polimorfismo.

- Jerarquías de clase.
- Herencia y su relación con reutilización de clases preexistentes.
- Control de acceso a los miembros de clase utilizando herencia.
- Clases abstractas.
- Polimorfismo.

Unidad III. Componentes básicos de interfaz gráfica.

- Diseño de ventanas para captura de datos y muestra de información.
- Objetivo de los diferentes objetos gráficos
- Diseño de objetos gráficos (cajas de texto, botones de comando, etiquetas, listas desplegables, botones de opción, cajas de chequeo, etc.)

Unidad IV. Programación por eventos.

- Modelo básico de programación por eventos
- Manejo de eventos del teclado y del "mouse"

Unidad V. Programación en capas.

- Modelo básico de programación en capas:
 - Capa de vista
 - Capa de lógica de negocios
 - Capa de Base de Datos

Unidad VI. Excepciones.

- Validaciones de entrada de datos
- Manejo de excepciones.

Unidad VII. Memoria dinámica

- Listas simplemente enlazadas
 - Operaciones básicas con listas
- Listas doblemente enlazadas
 - Operaciones básicas con listas doblemente enlazadas.

Unidad VIII. Estructuras de datos interconstruidas

- Uso y aplicación de Registros de información,
- Colas: Operaciones sobre colas
 - Implementación sobre arreglos
 - Implementación sobre listas
- Uso y aplicación de Árboles binarios y n-arios.
 - Implementación sobre arreglos
 - Implementación sobre listas

Metodología

El curso se lleva a cabo en forma teórico-práctica con uso de laboratorio y con una amplia participación por parte del estudiante y trabajo extra clase.

Evaluación

Examen I	15%
Examen II	15%
Prácticas en Clase	15%
I Proyecto	25%
II Proyecto	25%
Quices y Tareas	5%
Total	100%

Bibliografía

Arnou, D. y Weiss, G. (2001). **Introducción a la programación con Java**. Addison Wesley.

Deitel y Deitel.(1998). **Cómo programar en Java**. Prentice Hall.

Humphrey, Watts S.(1997). **Introduction to the Personal Software Process**.Addison Wesley.

Larman, Craig. (1999). **UML y Patrones: Introducción al análisis y diseño orientado a objetos**. Prentice Hall.

Lemay, L. y Cadenhead, R. (1999). **Aprendiendo Java 2 en 21 días**.Prentice Hall.

Meyer, B. (1997). **Object-Oriented Software Construction**.Prentice Hall.

Page-Jones, M. (2000).**Fundamentals of Object-Oriented Design in UML**. Addison Wesley.

Savitch, W. J. (1999). **Java: an introduction to computer science and programming**. Prentice Hall.

Cronograma

Semana	Lección	Unidad	Actividad
Semana 1	1	GitLab, Orientación a objetos (Relaciones entre objetos)	Diagnostico
Semana 2	2	Orientación a objetos(Relaciones entre objetos)	Clase magistral, práctica individual
Semana 3	3	Orientación a objetos (Herencia y Polimorfismo)	Clase magistral
Semana 4	4	Interfaz gráfica, Programación por eventos, Excepciones	Práctica individual
Semana 5	5	Estructura de datos (Memoria Dinámica)	Clase magistral. Primer proyecto programado
Semana 6	6	Estructura de datos (Memoria Dinámica)	Clase magistral, práctica individual
Semana 7	7	Estructura de datos (Operaciones en Colas y Listas)	Práctica individual
Semana 8	8	Programación por capas	Clase magistral
Semana 9	9	Revisión del Primer Proyecto	Segundo proyecto programado
Semana 10	10	Integración con Motor de Base de Datos	Clase magistral
Semana 11	11	Integración con Motor de Base de Datos	Clase magistral, práctica individual
Semana 12	12	Integración con Motor de Base de Datos	Práctica individual
Semana 13	13	Examen Final	
Semana 14	14	Revisión del Segundo Proyecto	