

Datos Generales

- maikol.guzman@avantica.net
- mikeguzman@gmail.com

Paradigma de Programación

"Un paradigma de programación es una propuesta tecnológica que es adoptada por una comunidad de programadores cuyo núcleo central es incuestionable en cuanto a que únicamente trata de resolver uno o varios problemas claramente delimitados."

Curso

Créditos	Horas semanales	Horas presenciales (pueden variar según los contenidos específicos)		Horas de estudio independiente
4	11	Teoría	Práctica	7
		3(2)	1(2)	

Objetivos

- Panorama general sobre los principales paradigmas de programación.
- Técnicas que existen.



Evaluación



Primer Examen Parcial	20%
Segundo Examen Parcial	20%
Quices y Tareas	10%
Trabajo de Investigación (I y II)	20%
Proyectos (I y II)	30%

Grupos de trabajo



Proyectos

Trabajos de Investigación y Presentaciones

Contenidos

Parte 1 – Características generales de los lenguajes de programación

1. *Introducción a los lenguajes de programación (Objetivos 1 ,2, 4)*
 - a. Historia de los lenguajes de programación.
2. *Traducción y compilación*
 - a. Autómatas de estado Finito (FSA)
 - b. Gramáticas y definición formal de lenguajes de programación.
 - c. Analizadores sintácticos y generación de código.
3. *Lenguajes de programación y arquitectura (Objetivos 2, 4)*
 - a. Tipos de datos y mecanismos de abstracción.
 - b. Métodos de encapsulamiento.
 - c. Paradigmas de programación.
 - i. Modelos computacionales teóricos
 - ii. Programación imperativa
 - iii. Programación declarativa
 1. Programación funcional
 2. Programación lógica
 - iv. Otros paradigmas de programación

Contenidos

Parte 2 – Programación funcional

4. Programación Funcional (Objetivos 3, 5, 7)

- a. Introducción al cálculo lambda (λ -calculus)
- b. Recursividad y funciones recursivas primitivas.
- c. Recursividad simple, recursividad lineal y de cola.
- d. Limitaciones de la programación funcional.
- e. El problema del reconocimiento de patrones
- f. Expresiones regulares y Evaluación de expresiones

Parte 3 – Programación lógica

5. Programación lógica (Objetivos 2,3,4,7)

- a. Programación lógica.
- b. Principios teóricos de la programación lógica.
- c. Lógica cuantificada de primer orden.
- d. El algoritmo de unificación y el método de resolución.
- e. Declaraciones recursivas.

Contenidos

Parte 4 – Programación imperativa y Programación Orientada a Objetos

6. Programación imperativa (Objetivos 2, 3, 5)

- a. Las estructuras básicas y la programación estructurada por control.
- b. Comparación entre la programación imperativa secuencial y la programación algorítmica (estructurada).

7. Programación guiada por eventos (Objetivos 2, 3, 6)

- a. Programación guiada por eventos (*event-driven programming*) y programación guiada por flujo (*flow-driven programming*).
- b. Modelo básico de la programación guiada por eventos.
- c. El problema del manejo de la estructura de control básica (ciclo de atención de eventos).
- d. Relación entre la implementación de las interfaces gráficas de usuario (GUIs) y la arquitectura de una aplicación de software.
- e. Serialización de interfaces de usuario.

8. Programación orientada a objetos (Objetivos 2, 6)

- a. Conceptos fundamentales y propósito de la POO (Programación Orientada a Objetos).
- b. Principios empíricos de la POO.
- c. Ventajas y desventajas de la POO. Consecuencias del empleo de técnicas de POO en el desarrollo de sistemas de información.

Lineamientos para proyectos

- Orden en el código
- Documentación
- Revisar estándares de programación
- Leer siempre las instrucciones

Buenas Prácticas de programación

