

Nombre de la asignatura: **TEMAS AVANZADOS DE PROGRAMACIÓN**

Línea de Trabajo: **Ingeniería de Software**

Tiempo de dedicación del estudiante a las actividades de:

DOC	TIS	TPS	Horas Totales	Créditos
48	20	100	168	6

**DOC:** Docencia; **TIS:** Trabajo independiente significativo; **TPS:** Trabajo profesional supervisado

### 1. Historial de la asignatura

Fecha revisión/actualización	Participantes	Observaciones, cambios o justificación
21/Noviembre/11	MTI Lilia del Carmen García Mundo MC. Juan Antonio Vargas Enríquez  MSI María Gabriela Salazar de León	Versión preliminar

### 2. Pre-requisitos y Correquisitos

Haber programado en algún lenguaje de programación orientado a objetos.

### 3. Objetivo de la asignatura

Aplicar los mecanismos de programación con genéricos, meta programación, reutilización mediante mixins y envoltentes, así como de programación generativa para el desarrollo de aplicaciones reconfigurables a tiempo de ejecución.

### 4. Aportación al perfil del graduado

El alumno será capaz de:

- Diseñar aplicaciones con código reutilizable para resolver problemas del sector productivo público y privado.

### 5. Contenido temático

Unidad	Temas	Subtemas
1	Programación con genéricos y polimorfismo con genéricos.	1. Introducción 2. Métodos genéricos a. Implementación b. Traducción en tiempo de compilación 3. Clases genéricas

2	Meta-programación	<ol style="list-style-type: none"> <li>1. Que es la meta programación?</li> <li>2. Manipulación de tipos y características <ol style="list-style-type: none"> <li>a. Asociaciones de tipos</li> <li>b. Metafunciones</li> <li>c. Metafunciones numéricas</li> <li>d. Toma de decisiones en tiempo de compilación</li> </ol> </li> </ol>
3	Polimorfismo en familias de tipos: mixins y envolventes	<ol style="list-style-type: none"> <li>1. Introducción a la programación en Ruby</li> <li>2. Estructura y ejecución de programas en Ruby</li> <li>3. Tipos de datos y objetos</li> <li>4. Metodos, procs, lambdas</li> <li>5. Clases y módulos <ol style="list-style-type: none"> <li>5.1 Implementación de Mixins</li> </ol> </li> <li>6. Metaprogramación</li> </ol>
4	Programación generativa	<ol style="list-style-type: none"> <li>1. Ingeniería de Dominio</li> <li>2. Modelado de Características</li> <li>3. Programación genérica</li> <li>4. Programación orientada a aspectos</li> <li>5. Metaprogramacion de plantillas</li> <li>6. Generadores</li> </ol>

## 6. Metodología de desarrollo del curso

El profesor expondrá los conceptos teóricos y ejemplos de los contenidos del curso. El alumno realizará ejercicios solicitados por el profesor y construirá un proyecto integrador que incluya la aplicación práctica de cada uno de los temas del curso.

## 7. Sugerencias de evaluación

Exámenes teórico-prácticos que permitan evaluar la asimilación de los conceptos 30%

Desarrollo de ejercicios prácticos 20%

Proyecto Integrador 50%

## 8. Bibliografía y software de Apoyo

C++ Template Metaprogramming Concepts, Tools and Techniques from Boost to Beyond

David Abrahams and Aleksey Gurtovoy. Addison-Wesley , 2004

The Ruby Programming Language. David Flanagan and Yukihiro Matsumoto. O'Reilly Media, 2008

Generative Programming: Methods, Tools, and Applications. Krzysztof Czarnecki and Ulrich Eisenecker. Addison-Wesley , 2000

## 9. Actividades propuestas

Unidad	Actividades
1	Desarrollo de una aplicación que implemente clases genéricas en Java
3	Desarrollo de aplicación simple que implemente mixins y envolventes en Ruby o C++

## 10. Catedrático (s) responsable (s)

MTI. Lilia del Carmen García Mundo

