

Ampliación de Programación

- **Características**
 - 6 ECTS, primer semestre
 - Semanalmente: 3 horas teoría, 2 horas prácticas
- **Objetivos**
 - Fortalecer conocimientos de programación mediante elementos de programación funcional en el contexto de un lenguaje funcional puro (*Haskell*). Comparación con otros paradigmas de programación.
 - Ampliar conocimientos de Programación Orientada a Objetos (POO), incorporando componentes de programación funcional en el contexto del entorno de desarrollo .NET. Formulación de patrones de diseño de software.
- **Requisitos previos**
 - Programación en *C*, *Java*, *Lisp*
- **Interés académico**
 - Estudiar un paradigma de programación alternativo y de uso general.
 - Ampliar conocimientos de POO usando el entorno integrado .NET que añade componentes funcionales a un entorno de programación imperativa, y que permite plantear los patrones de diseño de acuerdo a esas componentes.

Ampliación de Programación

- **Programa** resumido
 - 50-60% .Programación en *Haskell* (*open source*, lenguaje funcional puro). Común en entornos más académicos.
 - Transparencia referencial, evaluación perezosa, mónadas, etc
 - 50-40% . Desarrollo en *.NET* (propietario), usando los lenguajes *C#* (más imperativo) y *F#* (más funcional). Uso de POO y de elementos funcionales. Delegados, eventos, genéricos, etc. Patrones de diseño. Muy usado en empresas.
- **Metodología** docente
 - Teoría (aula y laboratorio)
 - Prácticas (laboratorio, entregas de pequeños proyectos)
- **Evaluación**
 - Teoría (70%). Prácticas (30%).
 - Evaluación continua: 3 parciales (25%, 35%, 40%)
 - Evaluación no continua: examen final
- **Profesor** coordinador: Francisco Saiz