

**FACULTAD** : CIENCIAS DE LA INGENIERÍA  
**CARRERA** : INGENIERÍA CIVIL INFORMÁTICA  
**CURSO** : MÉTODOS FORMALES  
**CRÉDITOS** : 10  
**CÓDIGO** : ICI-521  
**REQUISITOS** : VIII SEMESTRE APROBADO.

## **I.- DESCRIPCIÓN O FUNDAMENTACIÓN DE LA ASIGNATURA**

En el curso se analizarán los elementos teóricos fundamentales de formalismos aplicados en la construcción de software, desde el punto de vista estrictamente formal, de ciencias de la computación e ingeniería de software. Esta asignatura relaciona los conocimientos matemáticos aplicados en la especificación de software para especificar soluciones aplicadas en la línea de programación e ingeniería del software.

## **II.- OBJETIVO GENERAL.**

Especificar Software Complejo utilizando métodos formales.

## **III.- OBJETIVOS ESPECÍFICOS.**

- Aplicar los principales métodos formales de la ingeniería de software, las capacidades y limitaciones intrínsecas del uso de los formalismos en el desarrollo de sistemas.
- Categorizar la aplicabilidad de los diferentes métodos formales.
- Especificar formalmente las interfaces y el comportamiento de software utilizando lenguaje matemático riguroso.

## **IV.- CONTENIDOS**

### **A.- Introducción**

Qué son los métodos formales  
Grado de maduración de la tecnología de métodos formales  
Rol del formalismo en las distintas etapas del ciclo de vida  
Uso de métodos formales versus desarrollo formal.

### **B.- Lógica de Primer Orden**

Sintaxis, Semántica, Interpretaciones.

### **C.- Lenguajes de Especificación Algebraica**

Especificación formal de tipos de datos  
Especificación formal de tipos abstractos de datos

### **D.- Lenguajes Rigurosos de Especificación de Software**

Uso de formalismos para la especificación

Especificación de comportamiento de sistemas

## **V.- METODOLOGÍA**

Clases teórico-expositivas apoyada con presentaciones proporcionadas por el Docente y apoyadas por material Bibliográfico del curso. Clases de desarrollo práctico de ejercicios, donde los alumnos se enfrentarán a la realización de problemas reales de especificación de software. Desarrollo de investigación personal y resolución de tareas.

## **VI.- EVALUACIÓN**

Con el objeto de verificar el manejo conceptual y procedimental de los objetivos, se realizará pruebas escritas, cuyo promedio tendrá una ponderación del 50 % y una prueba acumulativa final equivalente al 30%.

Con el objeto de verificar el correcto uso de los formalismos aplicados al diseño de programas, se desarrollarán controles y tareas evaluadas. El promedio de estas tareas equivale al 20%.

## **VII.- BIBLIOGRAFÍA**

### **BÁSICA**

Bertossi Leopoldo (1996), *Lógica Para Ciencia de la Computación*, Ediciones Universidad Católica de Chile, Santiago, Chile.

Bjørner, Dines; Henson, Martin C (2007), *Logics of Specification Languages*, Springer, Germany.

Brassard, G., P. Bratley (2000), *Fundamentos de algoritmia*, Pearson, España.

Cormen, Thomas H., Leiserson, Charles E., Rivest, Ronald L., Stein, Clifford (2001), *Introduction to Algorithms*, The MIT Press, 2nd Edition, U.S.A.

Grassman W. K. (1998), *Matemática Discreta y Lógica*, Prentice Hall, Madrid, España.

### **COMPLEMENTARIA**

Alagar. Vangalur (1989), *Fundamentals of Computing, Theory and Practice*, Ed. Prentice Hall, U.S.A.

Enderton Herbert (1973). *Introduction To Discrete Structures for Computer Science and Engineering*, Ed. Addison-Wesley, U.S.A.

Gordon Michael. (1988), *Programming Language Theory and Its Implementation*. Ed. Prentice Hall, U.S.A.

Harary Frank (1995), *Graph Theory*, Ed. Addison-Wesley, Massachussets, U.S.A.

Smith, Graeme (1999), *The Object-Z Specification Language*, Vol. 1, Springer, U.S.A.