

Universidad de Santiago de Chile  
Facultad de Ciencia  
Departamento de Matemática y Ciencia de la Computación.

## LICENCIATURA EN CIENCIA DE LA COMPUTACIÓN PROGRAMA DE ASIGNATURA

### Lenguajes y Técnicas de Programación

Nivel 2 - TEL : 4-0-2

#### I. Objetivos

Los alumnos que aprueben este curso deberán ser capaces de :

Conocer los conceptos de definición e implementación de los lenguajes de programación convencionales y desarrollar aplicaciones utilizando lenguajes que contribuyan a la identificación y comprensión de esos conceptos.

El Alumno resolverá algoritmos directamente traducibles a lenguajes de programación imperativa, utilizando las técnicas de programación estructurada y refinación progresiva.

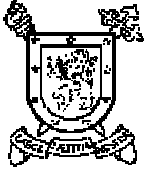
#### II. Contenidos

##### UNIDAD 1. Introducción.

- Evolución de los Conceptos
- Clasificación de los lenguajes de Programación
- Categorización de los Lenguajes de Programación
- Criterios de definición y diseño de lenguajes
- Sintaxis de lenguajes
- Gramática
- Metalenguaje BNF
- Metalenguaje Diagramas sintácticos
- Metalenguaje CBL
- Semántica de lenguajes
- Procesadores de lenguajes
- Elementos del Lenguaje C

##### UNIDAD 2. Objetos de Datos

- Componentes de un objeto
- Constantes
- Variables
- Declaraciones
- Elementos de lenguaje C



Universidad de Santiago de Chile  
Facultad de Ciencia  
Departamento de Matemática y Ciencia de la Computación.

- Binding
- Operaciones
- Expresiones

### **UNIDAD 3.** Sentencias

- Asignación
- Elementos de lenguaje C
- Inicialización
- Control de secuencia
- Desarrollo de ejercicios

### **UNIDAD 4** Tipos de Datos.

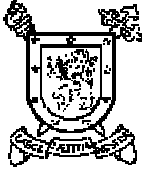
- Definiciones
- Estructura de tipos
- Comprobación de tipos
- Producto cartesiano
- Elementos de lenguaje C
- Aplicación finita
- Descriptores
- Fórmulas de acceso
- Unión discriminada
- Conjunto potencia

### **UNIDAD 5.** Parametrización Introducción.

- Importancia de la parametrización
- Llamada por nombre
- Llamada por referencia
- Llamada por copia
- Llamada por indirección
- Parametrización de tipos
- Parametrización de subprogramas

### **UNIDAD 6.** Control de Datos.

- Alcance estático de datos
- Alcance dinámico de datos
- Extensión de datos



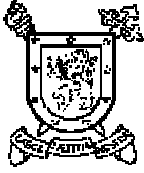
- Extensión de datos
- Elementos de lenguaje C
- Memoria heap
- Elementos de tamaño fijo
- Elementos de tamaño variable
- Síntesis del curso
- Desarrollo de ejercicios

## **UNIDAD 7. Calidad del Software y Modularidad**

- Introducción
  - Factores internos y externos de calidad.
  - Ciclo de vida y calidad del software.
  - Fiabilidad y corrección.
  - Documentación
- Modularidad
  - Ideas básicas y notación.
  - Aspectos de la calidad de un diseño modular: Cohesión y Acoplamiento.
  - Guías de diseño: Balanceado y factorización.
- Prueba de programas.
  - Principios de prueba.
  - Pruebas indirectas.
  - Pruebas directas: estrategias de caja blanca y caja negra.
  - Pruebas de nivel superior.
  - Depuración.

## **UNIDAD 8. Introducción a la OO**

- Limitaciones del diseño modular
  - Reutilización del software. Propuestas.
  - La propuesta de la orientación a objetos. (OMT-UML)
- La estructura estática: Clases.
  - El papel de las clases en orientación a objetos.
  - Un sistema de tipos uniforme.
  - El estilo de programación orientado a objetos.
  - Ocultación de la información.
- La estructura dinámica: Objetos.
  - Objetos.
  - Objetos y referencias.
  - Procedimientos de creación.
  - Operaciones sobre referencias.



- Objetos compuestos y tipos expandidos.
- Un ejemplo especial: la clase STRING.
- Ligadura por referencia y por valor.
- Trabajando con referencias, ventajas e inconvenientes.
- La gestión de la memoria.
- Genericidad y Herencia
  - Generalización horizontal y vertical.
  - La necesidad de parametrización de tipos.
  - Clases genéricas.
  - El mecanismo de la herencia

### **UNIDAD 9 Programación bajo contrato**

- Verificación de algoritmos.
  - Reglas de verificación.
  - Corrección parcial de un algoritmo.
  - Corrección total.
  - Algoritmos recursivos.

### **III. Metodología**

El curso se desarrollará preferentemente en base clases expositivas y actividades de ejercicio, a realizarse en clases colectiva o individualmente.

Conforme a disponibilidad y pertinencia, algunas sesiones presenciales se realizarán en el laboratorio y podrán apoyarse en presentaciones multimediales.

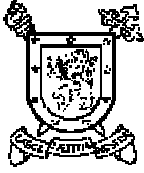
Las presentaciones, trabajos y parte del material bibliográfico estará disponible a través de Internet.

### **IV. Evaluación**

Se efectuarán dos (2) pruebas escritas durante el Semestre. Se programaran evaluaciones tareas individuales y grupales las que serán evaluadas en Laboratorio

### **V. Bibliografía.**

Programación Estructurada en lenguaje C .  
*LÓPEZ, Leobardo*



Universidad de Santiago de Chile  
Facultad de Ciencia  
Departamento de Matemática y Ciencia de la Computación.

**ISBN 970-15-1062-3**

*Alfaomega*

Programación Orientada a Objetos con C++ , 3ª Ed.

*CEBALLOS, Fco. Javier*

**ISBN 970-15-0956-0**

*Coedición: Alfaomega-Rama.*

Lenguaje y Aplicaciones .

*CEBALLOS, Fco. Javier*

**ISBN 970-15-1094-1**

*Coedición: Alfaomega-Rama*

Concepts, Techniques, and Models of Programming

*Peter Van Roy and Seif Haridi*

**ISBN ISBN 0-262-2206**

*Miti press, 9-5, available March 2004*