



Universidad Nacional de Lanús

Departamento de Desarrollo Productivo y Tecnológico

**Carrera: *Licenciatura en Sistemas***

**Asignatura: *CONCEPTOS y PARADIGMAS  
de LENGUAJES de PROGRAMACIÓN - Plan  
2011***

**Docente responsable: *Ing. Elida Leoni***

**Año: *2024***

**Cuatrimestre: *3º Año – 1º Cuatrimestre***

## ***1- Fundamentación de la Asignatura:***

Esta asignatura provee al estudiante de los conocimientos básicos necesarios que les posibiliten su introducción a los aspectos conceptuales de la problemática del software y sus paradigmas. La incorporación de estos aspectos reviste fundamental importancia para la actividad de desarrollo de software.

El estudio de los aspectos esenciales que conforman los diversos paradigmas permite al estudiante seleccionar la mejor forma para abordar un problema particular y propender a su solución.

La consideración de esta diversidad de conceptos y paradigmas preparan y predisponen al estudiante a ampliar el aprendizaje de los lenguajes de programación, disponiendo de una batería de herramientas que le permitirá ubicar estos conceptos dentro de cada uno de ellos y así facilitar su comprensión.

Dentro de la carrera y en esta materia, se infunde la comunicación efectiva para el correcto desarrollo, tanto como la acción ética y responsable en sus trabajos y pasos por la misma.

.

## ***2- Objetivos:***

- Acercar al estudiante a los diferentes conceptos y paradigmas de lenguajes de programación. Reforzar los conceptos de los lenguajes de programación.
- Que los estudiantes manejen fluidamente los paradigmas más utilizados.
- Que los estudiantes tengan los elementos conceptuales necesarios para poder definir qué paradigma se alinea al desarrollo por realizar.

## ***3- Contenidos Mínimos:***

Sintaxis y semántica.

Semántica operacional.

Entidades y ligaduras.

Sistemas de tipos.

Encapsulamiento y abstracción.

Intérpretes y Compiladores.

Paradigmas de lenguajes (imperativo, orientado a objetos, funcional, lógico).

## ***4- Contenidos:***

## **UNIDAD 1: SINTAXIS Y SEMÁNTICA.**

Repaso de conceptos básicos de la sintaxis y semántica de los diferentes paradigmas.

### Bibliografía:

Louden, K. C., & Lambert, K. A. (2011). Chapter 1: introduction. En *Programming languages: Principles and practices* (3.<sup>a</sup> ed.) (pp. 1–23). Cengage Learning.

[https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20\(3ed.%2C%20Course%20Technology%2C%202011\)\(ISBN%209781111529413\)\(666s\)\\_CsPl\\_.pdf](https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20(3ed.%2C%20Course%20Technology%2C%202011)(ISBN%209781111529413)(666s)_CsPl_.pdf)

Sebesta, R. E. (2012). Chapter 2: evolution of the major programming languages. En *Concepts of programming languages* (10.<sup>a</sup> ed.) (pp. 35–111). Pearson.

<https://www.ime.usp.br/~alvaroma/ucsp/proglang/book.pdf>

Sebesta, R. E. (2012). Chapter 3: preliminaries. En *Concepts of programming languages* (10.<sup>a</sup> ed.) (pp. 113–166). Pearson. <https://www.ime.usp.br/~alvaroma/ucsp/proglang/book.pdf>

## **UNIDAD 2: SEMÁNTICA OPERACIONAL.**

Definición de semántica operacional, secuencia de pasos válidos.

### Bibliografía:

Louden, K. C., & Lambert, K. A. (2011). Chapter 6: syntax. En *Programming languages: Principles and practices* (3.<sup>a</sup> ed.) (pp. 203–254). Cengage Learning.

[https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20\(3ed.%2C%20Course%20Technology%2C%202011\)\(ISBN%209781111529413\)\(666s\)\\_CsPl\\_.pdf](https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20(3ed.%2C%20Course%20Technology%2C%202011)(ISBN%209781111529413)(666s)_CsPl_.pdf)

Sebesta, R. E. (2012). Chapter 4: lexical and syntax analysis. En *Concepts of programming languages* (10.<sup>a</sup> ed.) (pp. 167–201). Pearson.

<https://www.ime.usp.br/~alvaroma/ucsp/proglang/book.pdf>

## **UNIDAD 3: ENTIDADES Y LIGADURAS.**

Conceptos de ligaduras y entidades, alcance y visibilidad.

## Bibliografía:

Louden, K. C., & Lambert, K. A. (2011). Chapter 6: syntax. En *Programming languages: Principles and practices* (3.<sup>a</sup> ed.) (pp. 203–254). Cengage Learning.

[https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20\(3ed.%2C%20Course%20Technology%2C%202011\)\(ISBN%209781111529413\)\(666s\)\\_CsPl\\_.pdf](https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20(3ed.%2C%20Course%20Technology%2C%202011)(ISBN%209781111529413)(666s)_CsPl_.pdf)

Louden, K. C., & Lambert, K. A. (2011). Chapter 7: basic semantics. En *Programming languages: Principles and practices* (3.<sup>a</sup> ed.) (pp. 255–323). Cengage Learning.

[https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20\(3ed.%2C%20Course%20Technology%2C%202011\)\(ISBN%209781111529413\)\(666s\)\\_CsPl\\_.pdf](https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20(3ed.%2C%20Course%20Technology%2C%202011)(ISBN%209781111529413)(666s)_CsPl_.pdf)

Sebesta, R. E. (2012). Chapter 4: lexical and syntax analysis. En *Concepts of programming languages* (10.<sup>a</sup> ed.) (pp. 167–201). Pearson.

<https://www.ime.usp.br/~alvaroma/ucsp/proglang/book.pdf>

Sebesta, R. E. (2012). Chapter 5: abstract data types and encapsulation constructs. En *Concepts of programming languages* (10.<sup>a</sup> ed.) (pp. 203–241). Pearson.

<https://www.ime.usp.br/~alvaroma/ucsp/proglang/book.pdf>

## **UNIDAD 4: SISTEMAS DE TIPOS.**

Definición de tipos, clasificación, equivalencias.

## Bibliografía:

Louden, K. C., & Lambert, K. A. (2011). Chapter 8: data types. En *Programming languages: Principles and practices* (3.<sup>a</sup> ed.) (pp. 325–399). Cengage Learning.

[https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20\(3ed.%2C%20Course%20Technology%2C%202011\)\(ISBN%209781111529413\)\(666s\)\\_CsPl\\_.pdf](https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20(3ed.%2C%20Course%20Technology%2C%202011)(ISBN%209781111529413)(666s)_CsPl_.pdf)

## **UNIDAD 5: ENCAPSULAMIENTO Y ABSTRACCIÓN**

Descripción de abstracción de datos y tipos abstractos, encapsulamiento.

## Bibliografía:

Louden, K. C., & Lambert, K. A. (2011). Chapter 11: abstract types and encapsulation constructs. En *Programming languages: Principles and practices* (3.<sup>a</sup> ed.) (pp. 473–522). Cengage Learning.

[https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20\(3ed.%2C%20Course%20Technology%2C%202011\)\(ISBN%209781111529413\)\(666s\)\\_CsPl\\_.pdf](https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20(3ed.%2C%20Course%20Technology%2C%202011)(ISBN%209781111529413)(666s)_CsPl_.pdf)

Pratt, T. W., & Zelkowitz, M. V. (2000). Chapter 6: encapsulation. En *Programming languages: Design and implementation* (4.<sup>a</sup> ed.) (pp. 200–263). Prentice Hall.

Sebesta, R. E. (2012). Chapter 11: abstract data types and encapsulation constructs. En *Concepts of programming languages* (10.<sup>a</sup> ed.) (pp. 473–522). Pearson.

<https://www.ime.usp.br/~alvaroma/ucsp/proglang/book.pdf>

## **UNIDAD 6: INTÉRPRETES Y COMPILADORES**

Definición de intérprete y compilador, tipos, clasificación, estructura básica de compiladores.

### Bibliografía:

Sebesta, R. E. (2012). Chapter 1: preliminaries. En *Concepts of programming languages* (10.<sup>a</sup> ed.) (pp. 1–34). Pearson. <https://www.ime.usp.br/~alvaroma/ucsp/proglang/book.pdf>

## **UNIDAD 7: PARADIGMAS DE LENGUAJES**

Descripción de lenguajes: imperativo, orientado a objetos, funcional, lógico.

### Bibliografía:

Louden, K. C., & Lambert, K. A. (2011). Chapter 3: functional programming. En *Programming languages: Principles and practices* (3.<sup>a</sup> ed.) (pp. 47–101). Cengage Learning.

[https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20\(3ed.%2C%20Course%20Technology%2C%202011\)\(ISBN%209781111529413\)\(666s\)\\_CsPl\\_.pdf](https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20(3ed.%2C%20Course%20Technology%2C%202011)(ISBN%209781111529413)(666s)_CsPl_.pdf)

Louden, K. C., & Lambert, K. A. (2011). Chapter 4: logic programming. En *Programming languages: Principles and practices* (3.<sup>a</sup> ed.) (pp. 103–139). Cengage Learning.

[https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20\(3ed.%2C%20Course%20Technology%2C%202011\)\(ISBN%209781111529413\)\(666s\)\\_CsPl\\_.pdf](https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20(3ed.%2C%20Course%20Technology%2C%202011)(ISBN%209781111529413)(666s)_CsPl_.pdf)

[20K.A.%20Programming%20languages..%20principles%20and%20practice%20\(3ed.%2C%20Course%20Technology%2C%202011\)\(ISBN%209781111529413\)\(666s\)\\_CsPl\\_.pdf](https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20(3ed.%2C%20Course%20Technology%2C%202011)(ISBN%209781111529413)(666s)_CsPl_.pdf)

Louden, K. C., & Lambert, K. A. (2011). Chapter 5: object-oriented programming. En *Programming languages: Principles and practices* (3.<sup>a</sup> ed.) (pp. 141–202). Cengage Learning.

[https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20\(3ed.%2C%20Course%20Technology%2C%202011\)\(ISBN%209781111529413\)\(666s\)\\_CsPl\\_.pdf](https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20(3ed.%2C%20Course%20Technology%2C%202011)(ISBN%209781111529413)(666s)_CsPl_.pdf)

Sebesta, R. E. (2012). Chapter 2: evolution of the major programming languages. En *Concepts of programming languages* (10.<sup>a</sup> ed.) (pp. 35–111). Pearson.

<https://www.ime.usp.br/~alvaroma/ucsp/proglang/book.pdf>

Sebesta, R. E. (2012). Chapter 15: functional programming languages. En *Concepts of programming languages* (10.<sup>a</sup> ed.) (pp. 671–725). Pearson.

<https://www.ime.usp.br/~alvaroma/ucsp/proglang/book.pdf>

Sebesta, R. E. (2012). Chapter 16: logic programming languages. En *Concepts of programming languages* (10.<sup>a</sup> ed.) (pp. 727–761). Pearson.

<https://www.ime.usp.br/~alvaroma/ucsp/proglang/book.pdf>

## **5- Metodología de Trabajo:**

La asignatura se estructura bajo la modalidad de clases teórico – práctica, siguiendo los lineamientos fijados en los estándares de acreditación de la carrera.

Los contenidos teóricos están explicitados según la secuencia establecida en las unidades que conforman el programa. El desarrollo de las prácticas se efectúa tanto en aula, como con la utilización de equipamiento informático.

## **6- Desarrollo de Actividades Prácticas:**

Se desarrollará un trabajo Teórico/Práctico.

Este trabajo se basará en los paradigmas de programación vistos en la materia. En él se presentarán preguntas teóricas que se corresponden con el material presentado y una parte práctica que deberá ser resuelta con alguna de las implementaciones disponibles con herramientas libres.

La modalidad es generar grupos de 2 a 4 personas o de manera individual en caso de tratarse un curso constituido por menos de 10 estudiantes. La tutoría a cada uno de los grupos será realizada por parte del profesor.

## **7- Evaluación y Acreditación:**

El proceso evaluativo consta de una evaluación parcial que incluye el temario de las unidades de la asignatura. En caso de no aprobarse la misma, el estudiante contará con un recuperatorio. En caso de no aprobarse el examen, el estudiante quedará en la condición de libre, en caso de aprobarse el mismo, la nota será determinada por la nota del examen y la aprobación del trabajo práctico entregado. El resultado será la calificación de la cursada.

El trabajo práctico final grupal se corresponde al tiempo estimado de cuatro semanas de dedicación. El trabajo tendrá una aprobación grupal, pero será defendido individualmente en la fecha de final a elegir. La nota resultante de la defensa estará influida por la nota del trabajo práctico y constará como la nota del final oral.

## **8- Bibliografía:**

Louden, K. C., & Lambert, K. A. (2011). *Programming languages: Principles and practices* (3.<sup>a</sup> ed.). Cengage Learning.

[https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20\(3ed.%2C%20Course%20Technology%2C%202011\)\(ISBN%209781111529413\)\(666s\)\\_CsPl\\_.pdf](https://github.com/MethkupalliVasanth/Books/blob/master/Louden%20K.C.%2C%20Lambert%20K.A.%20Programming%20languages..%20principles%20and%20practice%20(3ed.%2C%20Course%20Technology%2C%202011)(ISBN%209781111529413)(666s)_CsPl_.pdf)

Pratt, T. W., & Zelkowitz, M. V. (2000). *Programming languages: Design and implementation* (4.<sup>a</sup> ed.). Prentice Hall.

Sebesta, R. E. (2012). *Concepts of programming languages* (10.<sup>a</sup> ed.). Pearson.

<https://www.ime.usp.br/~alvaroma/ucsp/proglang/book.pdf>