



**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**  
**Faculty of Electrical and Computer Engineering**  
**COURSE SYLLABUS**  
*Programming Languages*

**1. CODE AND NUMBER OF CREDITS**

<b>CODE</b>	FIEC01552	
<b>NUMBER OF CREDITS: 4</b>	<b>Theoretical: 4</b>	<b>Practical: 0</b>

**2. COURSE DESCRIPTION**

The fundamental concepts of design and implementation of programming languages are presented. The various language paradigms and implementation topics are emphasized. Students will develop projects with different paradigms and languages.

**3. PRE-REQUISITES AND CO-REQUISITES**

<b>PRE-REQUISITES</b>	FIEC04622 OBJECT-ORIENTED PROGRAMMING
<b>CO-REQUISITES</b>	

**4. CORE TEXT AND OTHER REQUIRED REFERENCES FOR THE TEACHING OF THE COURSE**

<b>CORE TEXT</b>	1. Sebesta. "Concepts of Programming Languages", 10th Edition, 2012, Addison-Wesley.
<b>REFERENCES</b>	1. Pratt. "Programming Languages: Design and Implementation", 4th Edition, 2000, Prentice Hall.

**5. COURSE LEARNING OUTCOMES**

At the end of the course, the student will be able to:

1. Know about syntax and semantics of programming languages.
2. Understand the role of hardware in the implementation of programming languages.
3. Understand the grammar of languages in order to compare their characteristics and choose the appropriate language for developing an application.
4. Learn about the internal structure of computers and understand their relationship to software design.
5. Write programs using languages from different paradigms, as to demonstrate their characteristics and similarities.
6. Research a related topic and write a technical report with appropriate references.
7. Do an oral presentation on a research topic.

**6. COURSE PROGRAM**

I. Introduction (2 sessions – 4 hours)	<ul style="list-style-type: none"><li>• Program introduction, course organization, grading policies.</li><li>• Reasons to study programming languages.</li></ul>
II. Architecture and Grammar of Translators (1 session – 2 hours)	<ul style="list-style-type: none"><li>• Objectives of programming languages.</li><li>• Design models of programming languages.</li><li>• Semantics of programming languages.</li></ul>
III. Hardware and Software Architecture	<ul style="list-style-type: none"><li>• Hardware architectures.</li><li>• Software Architectures.</li><li>• Virtual machines and other execution approaches.</li></ul>
IV. Formal Grammars (1 session – 2 hours)	<ul style="list-style-type: none"><li>• Translation steps.</li><li>• Grammar types.</li><li>• Backus Naur Form, Context-Free Grammars.</li></ul>
V. Finite-State Languages (2 sessions – 4 hours)	



- State diagrams.
- Finite state automatas.
- Regular grammars.
- Regular expressions.
- Recursive descent parser.
- VI. Encapsulation (2 sessions – 4 hours)
  - Basic data types.
  - Structured data.
- VII. Data Types (2 sessions – 4 hours)
  - Abstract data types.
  - Type equivalence.
- VIII. Inheritance (2 sessions – 4 hours)
  - Inheritance and derived classes.
  - Abstract classes and friends.
  - Polymorphism.
- IX. Expressions (1 session – 2 hours)
  - Expressions.
  - Basic Instructions.
- X. Subprogram Control (1 session – 2 hours)
  - Activation registry.
  - Parameter passing.
- XI. Storage Management (1 session – 2 hours)
  - Stack vs. Heap.
  - Garbage Collection.
- XII. Languages Taught (11 sessions – 22 hours)
  - Structured and Object Oriented.
  - Scripting.
  - Functional.

**7. WORKLOAD: THEORY/PRACTICE**

Two two-hour sessions per week.

**8. CONTRIBUTION OF THE COURSE TO THE EDUCATION OF THE STUDENT**

An overview of design and implementation topics is presented. There is an emphasis on object-oriented and functional paradigms.  
 Students recognize the importance of life-long learning and improving their communication skills.  
 This course covers topics on programming languages and computer organization.  
 Students analyze, design and develop software solutions based on engineering methodologies.  
 Three projects are implemented, using three different programming paradigms.  
 Groups of students will make an oral presentation based on a research topic. They will also present a written technical report on said research and provide appropriate citations.  
 During the analysis, design and implementation process, the students will behave ethically.

BASIC TRAINING	PROFESSIONAL TRAINING	SOCIAL SKILLS DEVELOPMENT
	X	

**9. THE RELATIONSHIP BETWEEN THE LEARNING OUTCOMES OF THE COURSE AND THE LEARNING OUTCOMES OF THE DEGREE PROGRAM**

LEARNING OUTCOMES OF THE DEGREE PROGRAM*	CONTRIBUTION (High, Medium, Low)	LEARNING OUTCOMES OF THE COURSE**	THE STUDENT MUST:
a) An ability to apply knowledge of computing	Medium	5	Algorithm planning an implementation, to solve engineering problems.



	and mathematics appropriate to the discipline.			
b)	An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution.	High	3,5	Design a solution to an engineering problem.
c)	An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs.	Medium	5	Evaluate, through language comparison, the design of a solution.
d)	An ability to function effectively on teams to accomplish a common goal.	Medium	6,7	Teamwork for: analysis, design and implementation of a solution; research.
e)	An understanding of professional, ethical, legal, security and social issues and responsibilities.	High	6,7	Act ethically. The courses' website provides guidelines.
f)	An ability to communicate effectively with a range of audiences.	High	6,7	Research and present a topic. This is a group activity.
g)	An ability to analyze the local and global impact of computing on individuals, organizations, and society.	Low	6	Understand the benefits of free and open source software.
h)	Recognition of the need for and an ability to engage in continuing professional development.	High	1,2,6	Learn to learn, especially in relation to programming languages and research.
i)	An ability to use current techniques, skills, and tools necessary for computing practice.	High	1,2,3,4,5	Work on projects with new programming languages and in-development versions.
j)	Ability to lead, manage and undertake projects.	Low	5	Lead a team during the development of the course projects.

#### 10. EVALUATION IN THE COURSE



Evaluation activities	
Exams	X
Tests	
Homework/tasks	X
Projects	X
Laboratory/Experiments	
Class participation	
Visits	
Other	X

#### 11. PERSON RESPONSIBLE FOR THE CREATION OF THE SYLLABUS AND THE DATE OF ITS CREATION

<b>Created by</b>	Javier Tibau, M.Sc.
<b>Date</b>	May 9, 2013



12. APPROVAL

ACADEMIC SECRETARY OF THE ACADEMIC DEPARTMENT	DIRECTOR OF TECHNICAL ACADEMIC SECRETARY
NAME: Sra. Leonor Caicedo G.	NAME: Ing. Marcos Mendoza V.
SIGNATURE: 	SIGNATURE:  ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
Date of approval by the Directive Council: 2013-334 2013-08-12	<b>Ing. Marcos Mendoza V.</b> DIRECTOR DE LA SECRETARÍA TÉCNICA ACADÉMICA

13. VALIDITY OF THE SYLLABUS

RESOLUTION OF THE POLYTECHNIC BOARD:	13-10-269
DATE:	2013-10-17