



UNIVERSIDADE LUSÍADA
LISBOA

Course Unit Program
Compilers

Academic Year 2026/2027

1. Organic Unit

Faculty of Business and Economic Sciences

2. Course

Computer Engineering

3. Cycle

1st

4. Course Unit

Compilers (L9023)

5. Scientific Area

6. Curricular Year

3rd

7. Type of Curricular Unit / Semester

2nd Semester



UNIVERSIDADE LUSÍADA
LISBOA

8. Class type and workload (work time)

Theoretical classes:

Practical classes:

Theoretical-Practical Classes 30,00

Tutorial Guidance: 20,00

9. ECTS Credits

6,00

10. Scientific Area Coordinator

11. Regent

Prof. Doutora MARIA INÊS VASCONCELLOS FURTADO

12. Assistants

13. Teaching Language

Portuguese

14. General Objectives

Explore the theoretical and practical foundations of compilers, from the theoretical basis of Turing Machines to the implementation of lexical and syntactic analyzers.

15. Specific Objectives

By the end of the course, students should be able to understand and apply the principles of Turing Machines, understand the structure and phases of a compiler, design and specify programming languages, and implement basic lexical and syntactic analyzers.

16. Skills to be acquired

Develop practical skills in the construction of lexical and syntactic analyzers. Understand the theoretical foundations of programming languages and compilers. Analyze and discuss the importance of Turing Machines in the theory of computing and computability.

17. Teaching Methodology

The course includes theoretical-practical classes presenting the syllabus content and proposing exercises or small assignments to be completed in class or through independent research. Tutorial sessions focus on solving exercises using computers equipped with the required tools. In the second part, each student individually develops a project with a lexical and syntactic analyzer.



UNIVERSIDADE LUSÍADA
LISBOA

18. Course Unit Program

- 1 Turing machines
 - 1.1 Introduction to Turing Machines
 - Definition, structure and basic operations.
 - 1.2 Church-Turing theorem
 - Implications and importance in computer theory.
2. The Structure of a Compiler
 - Overview of the phases of a compiler:
 - Lexical analysis, syntactic analysis, semantic analysis, code generation, optimization and assembly.
3. Programming languages
 - 3.1 Main characteristics of programming languages.
 - 3.2 Programming language classifications and paradigms.
4. Language Specification and Design
 - Formal definition of programming languages.
 - Methods for designing a programming language.
5. Phases of a Compiler
 - 5.1 Lexical analysis:
 - Role of lexical analysis in the compilation process.
 - 5.2 Building a lexical analyzer:
 - Use of deterministic finite automata (AFDs) and regular expressions.
6. Syntactic Analysis and Error Correction
 - 6.1 The role of syntactic analysis in the compiler.
 - 6.2 Building a syntactic analyzer:
 - Top-down and bottom-up parsing methods.
 - Treatment of syntactic errors.

19. Assessment Methods

In accordance with the Regulation on the Assessment of Knowledge and Competences in force, continuous assessment comprises three oral interventions distributed throughout the teaching period (factor 3), one of which is the defence of the final project; one written test at the end of the teaching period, lasting 90 minutes (factor 3); and one final individual or group project (factor 2).

20. Didactic Resources

Classes will be given, whenever possible, using visual presentations.

Students will be provided with illustrative examples of conventional and non-conventional models and problems, as well as supplementary information on the subjects taught.

21. Sustainability Objectives



UNIVERSIDADE LUSÍADA
LISBOA

SDG 4 - Quality Education: strengthens digital literacy and lexical/syntactic analysis skills essential for computer engineering training.

SDG 9 - Industry, Innovation and Infrastructure: fosters the development of efficient compilation tools and reliable digital infrastructures.

SDG 12 - Responsible Consumption and Production: promotes computational efficiency and reuse of software components.

22. Keywords

Compilers

Formal Languages

Turing Machine



UNIVERSIDADE LUSÍADA
LISBOA

23. Main Bibliography

Author(s): Siek, J.G.
Title: Essentials of Compilation: An Incremental Approach in Python
Edition: 2 Year: 2023
Local:
Publisher: MIT Press
ISBN Code: 978-0262048249

Author(s): Thain, D.
Title: Introduction to Compilers and Language Design
Edition: 2 Year: 2023
Local: USA
Publisher: Notre Dame University
ISBN Code: 979-8655180260

Author(s): R. Sethi A. V. Aho and J. D. Ullman
Title: Compiladores Principios, Tecnicas E Ferramentas
Edition: Year: 2011
Local:
Publisher: Pearson
ISBN Code: 978-8588639249

24. Complementary Bibliography

Author(s): Rodger, S., Finley, T.
Title: JFLAP Version 7.1
Edition: Year: 2018
Local: On-line
Publisher: <https://www.jflap.org/>
ISBN Code:

Author(s): Sipser, M.
Title: Introduction to the Theory of Computation
Edition: 3rd Year: 2015
Local:
Publisher: Cengage Learning
ISBN Code: 978-1133187813



UNIVERSIDADE LUSÍADA
LISBOA

Ficha técnica MIP / MIP Datasheet	
Título / Title	Course Unit Program 'COMPILADORES': Academic Year 2026/2027
Autor / Author	Prof. Doutora MARIA INÊS VASCONCELLOS FURTADO
Classificação / Classification	-
Descritores / Descriptors	-
Colaborador(es) / Colaborator(s)	-
Data Hora / Date Time	
Formato de dados / Data Format	Texto / Text, PDF
Estatuto de utilização / Usage	Acesso público / Public Access
Relação / Relation	Version 1,0
(c) Universidade Lusíada, 2026	