

---

**Ano Letivo** 2019-20

---

**Unidade Curricular** LÓGICA E COMPUTAÇÃO

---

**Cursos** ENGENHARIA INFORMÁTICA (1.º ciclo)

---

**Unidade Orgânica** Faculdade de Ciências e Tecnologia

---

**Código da Unidade Curricular** 14781053

---

**Área Científica** CIÊNCIA DE COMPUTADORES

---

**Sigla**

---

**Línguas de Aprendizagem** Português-PT

---

**Modalidade de ensino** Presencial

---

**Docente Responsável** Daniel da Silva Graça

---

DOCENTE	TIPO DE AULA	TURMAS	TOTAL HORAS DE CONTACTO (*)
Daniel da Silva Graça	T; TP	T1; TP1	30T; 30TP
Diana Ferreira Rodelo	TP	TP2	30TP

\* Para turmas lecionadas conjuntamente, apenas é contabilizada a carga horária de uma delas.

ANO	PERÍODO DE FUNCIONAMENTO*	HORAS DE CONTACTO	HORAS TOTAIS DE TRABALHO	ECTS
2º	S2	30T; 30TP	168	6

\* A-Anual;S-Semestral;Q-Quadrimestral;T-Trimestral

### Precedências

Sem precedências

### Conhecimentos Prévios recomendados

Recomenda-se que o aluno tenha conhecimentos básicos de Matemática (incluindo noções elementares de teoria de conjuntos, de grafos, e de funções), assim como conhecimentos básicos de programação.

### Objetivos de aprendizagem (conhecimentos, aptidões e competências)

Após concluir esta unidade curricular, os alunos deverão ser capazes de:

- Converter expressões lógicas informais em fórmulas do cálculo proposicional e do cálculo de predicados.
- Determinar a validade de uma fórmula lógica, utilizando uma abordagem semântica.
- Utilizar regras de inferência para construir demonstrações no cálculo proposicional e de predicados.
- Obter fórmulas em forma normal disjuntiva/conjuntiva/prenexa a partir uma fórmula inicial dada.
- Discutir o conceito de máquina de Turing e identificar algumas das suas variantes.
- Conceber máquinas de Turing que decidam uma linguagem dada.
- Explicar o que são problemas não-computáveis e a sua significância, e identificar o Problema da Paragem como sendo um problema não-computável.
- Definir as classes P e NP e discutir a importância do problema "P=NP?".

### Conteúdos programáticos

- 1) Cálculo Proposicional: sintaxe, semântica, sistemas dedutivos (dedução natural), resultados de correção e completude.
- 2) Lógica de primeira ordem (cálculo de predicados): sintaxe, semântica, sistemas dedutivos (dedução natural), resultados de correção e completude.
- 3) Introdução à Teoria da Computabilidade: máquinas de Turing, tese de Church-Turing, problemas não-computáveis.
- 4) Complexidade Computacional: definições básicas, as classes P e NP, o problema "P = NP?", problemas NP-completos.

### Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular

Esta cadeira pretende dar fundamentos sólidos de lógica e de teoria da computação aos alunos. Relativamente à parte de lógica, é lecionado o cálculo proposicional e a lógica de primeira ordem, que são as pedras basais da lógica. Estas lógicas são apresentadas utilizando quer uma abordagem semântica, quer uma abordagem baseada em sistemas dedutivos. Relativamente à teoria da computação, é apresentado a máquina de Turing, que é um modelo fulcral na teoria da computação, já que é a idealização do computador digital. A máquina de Turing é então estudada sob uma perspetiva de computabilidade, sendo mostrado que há problemas não-computáveis (isto é, que não podem ser resolvidos através de programas informáticos), e sob uma perspetiva de complexidade computacional, sendo introduzidas as importantes classes P e NP.

### **Metodologias de ensino (avaliação incluída)**

A unidade curricular será ministrada com recurso a aulas presenciais. Haverá aulas teóricas para apresentação dos conceitos teóricos, e aulas teórico-práticas onde será exemplificado a aplicação dessa teoria, através de exemplos e resolução de problemas. Será facultada uma sebenta de apoio especialmente concebida para a disciplina, contendo toda a teoria relevante, bem como exemplos resolvidos. Também será fornecida uma lista de problemas, alguns dos quais serão resolvidos nas aulas, e outros serão deixados para que os alunos os resolvam em casa.

A avaliação será efetuada por meio de exame, sendo realizados dois testes escritos durante o semestre. Os alunos cuja média dos testes for maior ou igual a 9.5 valores ficam imediatamente aprovados e dispensados de exame. Todos os restantes alunos, incluindo os que não tenham realizado nenhum dos testes, são admitidos a exame. Poderá ser exigida uma prova complementar aos alunos que obtenham uma nota superior a 16 valores nos testes/exame.

---

### **Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular**

Dado que esta unidade curricular versa sobre conteúdos essencialmente abstratos, estes serão ministrados através de aulas teóricas, onde o professor explicará a teoria, e irá esclarecer eventuais dúvidas que os alunos tenham. Por se tratar de conteúdos teóricos, será importante apresentar exemplos e resolver problemas, já que muitos alunos têm dificuldades em perceber conceitos teóricos sem exemplos. Isso será efetuado nas aulas teórico-práticas. Finalmente, será fornecida uma sebenta com toda a teoria e exemplos, para que os alunos possam estar nas aulas concentrados em perceber os conceitos apresentados, sem necessitarem de desviar a sua atenção das explicações para copiar do quadro definições, resultados, etc. Essa sebenta também permite que os alunos acompanhem a matéria caso não possam assistir ocasionalmente a alguma aula.

---

### **Bibliografia principal**

- [1] Sebenta fornecida pelo docente.
- [2] M. Ben-Ari. Mathematical Logic for Computer Science. 3rd edition. Springer, 2012.
- [3] M. Sipser. Introduction to the Theory of Computation, 3rd edition, Cengage Learning, 2012.

**Academic Year** 2019-20

**Course unit** LOGIC AND COMPUTATION

**Courses** INFORMATICS (COMPUTER SCIENCE) (1st Cycle)

**Faculty / School** FACULTY OF SCIENCES AND TECHNOLOGY

**Main Scientific Area** CIÊNCIA DE COMPUTADORES

**Acronym**

**Language of instruction** Portuguese-PT

**Teaching/Learning modality** Presential

**Coordinating teacher** Daniel da Silva Graça

Teaching staff	Type	Classes	Hours (*)
Daniel da Silva Graça	T; TP	T1; TP1	30T; 30TP
Diana Ferreira Rodelo	TP	TP2	30TP

\* For classes taught jointly, it is only accounted the workload of one.

### Contact hours

T	TP	PL	TC	S	E	OT	O	Total
30	30	0	0	0	0	0	0	168

T - Theoretical; TP - Theoretical and practical ; PL - Practical and laboratorial; TC - Field Work; S - Seminar; E - Training; OT - Tutorial; O - Other

### Pre-requisites

no pre-requisites

### Prior knowledge and skills

We recommend that students have a basic knowledge of mathematical concepts such as set theory, graphs, or functions, and an elementary knowledge of how to program.

### The students intended learning outcomes (knowledge, skills and competences)

At the end of this curricular unit, students should be able to:

- Convert logical statements from informal language to propositional and predicate logic expressions.
- Determine the validity of a logical formula, using a semantic approach.
- Use the rules of inference to construct proofs in propositional and predicate logic.
- To obtain normal disjunctive/conjunctive/prenex formulas from a given formula.
- Discuss the concept of Turing machines and to recognize some of its variants.
- Design Turing machines which decide a specified language.
- Explain what are non-computable problems and their significance, and to identify the Halting Problem as a non-computable problem.
- Define the classes P and NP and discuss the importance of the problem "P=NP?".

### Syllabus

- 1) Propositional Logic: syntax, semantics, deductive systems (natural deduction), soundness and completeness results.
- 2) First-order logic (predicate calculus): syntax, semantics, deductive systems (natural deduction), soundness and completeness results.
- 3) Introduction to the Theory of Computation: Turing machines, Church-Turing thesis, non-computable problems.
- 4) Computational Complexity: basic definitions, classes P and NP, the problem "P = NP?", NP-complete problems.

### Demonstration of the syllabus coherence with the curricular unit's learning objectives

The aim of this course is to provide a solid foundation in logic and in the theory of computation. Concerning logic, propositional logic and first-order logic are introduced, since they are the cornerstones of logic. These logics are presented using either a semantic approach or an approach based on deductive systems. Regarding the theory of computation, we present the Turing machine, which is a key model, since it is the idealization of the digital computer (by the Church-Turing thesis, also introduced in the course). Turing machines are then studied from a computability perspective. In particular, students will learn that there are non-computable problems (i.e., which cannot be solved by using computer programs). We also give fundamental notions of computational complexity, introducing the important classes P and NP and the problem "P=NP?".

### **Teaching methodologies (including evaluation)**

The course will be taught using lectures. There will be (theoretical) lectures to present the theoretical concepts and practical classes where the theory will be exemplified through examples and problem solving. Lecture notes (in Portuguese) specially designed to support the course and containing all the relevant theory and worked examples will be provided to the students. A list of problems will also be provided. Some of the problems will be solved in the classroom, and others will be left for the students to solve out of classroom.

The evaluation is done by a final exam. During the semester there will be two tests. All students with an average grade on the tests greater or equal to 9.5 values are immediately approved and exempt from the exam. All other students, including those who did not attend the tests, are admitted to the final exam. A complementary test may be performed for students having a grade superior to 16 values in the tests or exam.

---

### **Demonstration of the coherence between the teaching methodologies and the learning outcomes**

Since this course deals primarily with theoretical content, students will be taught through lectures where the teacher will explain the theory and will clarify any questions that students might have. Because of the theoretical nature of the subjects introduced in the classroom, it will be important to provide examples and solve problems, since many students have difficulties in understanding theoretical concepts without examples. This will be done in practical classes. Finally, lecture notes (in Portuguese) with all the theory and worked examples will be provided, so that students may focus on learning in the classroom, instead of constantly diverting their attention to take notes of definitions, results, etc. written on the blackboard. These lecture notes also allow students to follow the course if they are occasionally unable to attend one class.

---

### **Main Bibliography**

- [1] Lecture notes supplied by the teacher (in Portuguese).
- [2] M. Ben-Ari. Mathematical Logic for Computer Science. 3rd edition. Springer, 2012.
- [3] M. Sipser. Introduction to the Theory of Computation, 3rd edition, Cengage Learning, 2012.