

---

**Ano Letivo** 2020-21

---

**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 e/ou a distância.

---

**Docente Responsável** Diana Ferreira Rodelo

DOCENTE	TIPO DE AULA	TURMAS	TOTAL HORAS DE CONTACTO (*)
Diana Ferreira Rodelo	T; TP	T1; TP1; TP2	28T; 56TP

\* 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	28T; 28TP	156	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.

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

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.

Haverá duas frequências durante o semestre. Serão aprovados, e dispensados de exame, todos os alunos que obtenham nota igual ou superior a 6 valores em cada frequência e média igual ou superior a 9,5 valores nas duas frequências. Os restantes alunos poderão ser aprovados por exame com classificação igual ou superior a 9,5 valores. Qualquer aluno poderá ser sujeito a uma prova complementar, caso tenha ocorrido alguma irregularidade durante a sua avaliação.

---

### **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** 2020-21

---

**Course unit** LOGIC AND COMPUTATION

---

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

---

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

---

**Main Scientific Area**

---

**Acronym**

---

**Language of instruction** Portuguese-PT

---

**Teaching/Learning modality** Presential and/or online at distance.

---

**Coordinating teacher** Diana Ferreira Rodelo

---

Teaching staff	Type	Classes	Hours (*)
Diana Ferreira Rodelo	T; TP	T1; TP1; TP2	28T; 56TP

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

### Contact hours

T	TP	PL	TC	S	E	OT	O	Total
28	28	0	0	0	0	0	0	156

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.

### Teaching methodologies (including evaluation)

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.

There will be two tests. Students with grade equal or greater than 6 on each test and an average on both tests equal or greater than 9,5 will be approved and dispensed from the exam. The remaining students will be approved by exam with a classification equal or greater than 9,5. Students may be subject to a complimentary test to be approved, in case there was any kind of irregularity during their evaluation process.

### 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.