

---

**Ano Letivo** 2022-23

---

**Unidade Curricular** ANÁLISE NUMÉRICA I

---

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

---

**Unidade Orgânica** Faculdade de Ciências e Tecnologia

---

**Código da Unidade Curricular** 140064324

---

**Área Científica** MATEMÁTICA

---

**Sigla** MAT

---

**Código CNAEF (3 dígitos)** 461

---

**Contributo para os Objetivos de Desenvolvimento Sustentável - 4 ODS (Indicar até 3 objetivos)**

---

**Línguas de Aprendizagem** Português-PT

---

**Modalidade de ensino**

Presencial

---

**Docente Responsável**

Celestino António Maduro Coelho

---

DOCENTE	TIPO DE AULA	TURMAS	TOTAL HORAS DE CONTACTO (*)
Celestino António Maduro Coelho	T; TP	T1; TP1; TP2; TP3	28T; 84TP

\* 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	S1	28T; 28TP	156	6

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

---

**Precedências**

Sem precedências

---

**Conhecimentos Prévios recomendados**

É recomendável que o aluno possua bons conhecimentos de Análise Matemática, Álgebra Linear e Métodos de Programação.

---

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

Com a aprovação na disciplina, para além da apreensão dos fundamentos de Análise Numérica, o aluno deverá ser capaz de identificar questões importantes relativas aos tópicos estudados, em particular sobre estabilidade e condicionamento. Deverá também ser capaz de fazer uma escolha crítica de entre os algoritmos disponíveis para a resolução numérica de um dado problema.

### Conteúdos programáticos

1. Precisão finita
  2. Equações não lineares
  3. Sistemas de equações lineares
  4. Interpolação polinomial
  5. Integração numérica
- 

### Metodologias de ensino (avaliação incluída)

As aulas são divididas em teóricas (T) e teórico-práticas (TP). Nas aulas T são apresentados os conteúdos teóricos do programa e exemplos. Nas aulas TP são resolvidos problemas numéricos recorrendo à linguagem de programação Python.

A avaliação é realizada em dois modelos: avaliação contínua (AC) e avaliação por exame final (EF).

#### CA:

Este modelo de avaliação consiste na realização de dois testes (T1 e T2) com iguais pesos, 45% cada, e de um conjunto de trabalhos individuais, analíticos ou computacionais, (TI), com peso igual a 10%. Os trabalhos individuais computacionais são problemas numéricos para resolver utilizando o Python. A nota final na AC é definida por  $FAC=0.45(T1+T2)+0.1TI$ . Os alunos com  $FAC \geq 9.5$  ficam dispensados de exame.

#### EF:

O exame final (em todas as épocas) possui ponderação igual a 100%.

Durante o semestre os alunos podem utilizar o horário de tutoria semanal para esclarecer as dúvidas nos conteúdos lecionados e nos trabalhos individuais propostos.

---

### Bibliografia principal

- Heitor Pina, *Métodos Numéricos*, Escolar Editora, 2010
- Kendall E. Atkinson, *An Introduction to Numerical Analysis*, 2nd edition, Wiley, 2003
- Annette Burden, Douglas Faires & Richard Burden, *Numerical Analysis*, Cengage Learning, 2015
- Åke Björck, *Numerical Methods in Matrix Computations*, Springer, 2015

---

**Academic Year** 2022-23

---

**Course unit** NUMERICAL ANALYSIS I

---

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

---

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

---

**Main Scientific Area** MATH

---

**Acronym**

---

**CNAEF code (3 digits)** 461

---

**Contribution to Sustainable Development Goals - SGD (Designate up to 3 objectives)** 4

---

**Language of instruction** Portuguese

---

**Teaching/Learning modality** In class.

**Coordinating teacher** Celestino António Maduro Coelho

Teaching staff	Type	Classes	Hours (*)
Celestino António Maduro Coelho	T; TP	T1; TP1; TP2; TP3	28T; 84TP

\* 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

It is advisable that a student enrolled in this course have good skills in Mathematical Analysis, Linear Algebra and Programming Methods.

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

This unit should provide students with the basic features of numerical analysis. In particular they should be able to identify problems related to conditioning, stability and accuracy. Moreover, after successful completing this unit, students should be able to critically choose the right algorithm to solve, numerically, a given problem.

#### Syllabus

1. Finite precision
2. Nonlinear equations
3. Systems of linear equations
4. Polynomial interpolation
5. Numerical integration

### Teaching methodologies (including evaluation)

Classes are expository and include examples and exercises to apply the acquired knowledge.

The assessment comprises two models: continuous assessment (CA) and evaluation through final exam (FE).

#### CA:

This model of assessment comprises two mid-term exams (T1 and T2) with equal weights, 45% each, and a set of analytical or computational homework assignments (TI), 10% of the final CA grade. The computational homework assignments are numerical problems to solve using Python programming language. The final CA grade is given by  $FCA=0.45(T1+T2)+0.1TI$ . Students with  $FCA \geq 9.5$  are exempt from the final exam.

#### FE:

The final exam option consists of a single exam (100% of the final grade).

During the semester students may use tutorial time to clarify their difficulties in grasping the theory and in gaining practical knowledge, as well as in the development of the necessary skills for the computational assignments.

---

### Main Bibliography

- Heitor Pina, *Métodos Numéricos*, Escolar Editora, 2010
- Kendall E. Atkinson, *An Introduction to Numerical Analysis*, 2nd edition, Wiley, 2003
- Annette Burden, Douglas Faires & Richard Burden, *Numerical Analysis*, Cengage Learning, 2015
- Åke Björck, *Numerical Methods in Matrix Computations*, Springer, 2015