

---

**Ano Letivo** 2022-23

---

**Unidade Curricular** PROGRAMAÇÃO

---

**Cursos** ENGENHARIA DE SISTEMAS E TECNOLOGIAS INFORMÁTICAS (1.º ciclo)

---

**Unidade Orgânica** Instituto Superior de Engenharia

---

**Código da Unidade Curricular** 19411000

---

**Área Científica** CIÊNCIAS INFORMÁTICAS

---

**Sigla**

---

**Código CNAEF (3 dígitos)** 481

---

**Contributo para os Objetivos de Desenvolvimento Sustentável - 8; 9 ODS (Indicar até 3 objetivos)**

---

**Línguas de Aprendizagem** Português-PT.

**Modalidade de ensino**

Presencial.

**Docente Responsável**

Mário Alexandre Nobre Saleiro

| DOCENTE                       | TIPO DE AULA | TURMAS   | TOTAL HORAS DE CONTACTO (*) |
|-------------------------------|--------------|----------|-----------------------------|
| Mário Alexandre Nobre Saleiro | PL; TP       | TP1; PL1 | 14TP; 42PL                  |

\* 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 |
|-----|---------------------------|-------------------|--------------------------|------|
| 1º  | S1                        | 14TP; 42PL        | 130                      | 5    |

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

**Precedências**

Sem precedências

**Conhecimentos Prévios recomendados**

Não aplicável.

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

A disciplina de Programação tem por objectivo dotar os alunos com os conceitos básicos de programação procedimental em linguagens de alto nível.

Pretende-se que os alunos adquiram os conceitos indispensáveis à resolução algorítmica de problemas, com especial ênfase nos que surgem habitualmente na área dos sistemas computacionais, estruturação de aplicações, e abstracção procedimental e de dados.

A unidade curricular consiste na resolução de problemas tipificados e codificação numa linguagem de programação. Aplica-se inicialmente o paradigma procedimental, para posteriormente ser introduzida a programação orientada a objetos.

### Conteúdos programáticos

1. Introdução e Motivação
  2. Algoritmia
  3. Ambiente de desenvolvimento
  4. Variáveis e tipos de dados
  5. Entrada e saída de dados
  6. Estruturas de controlo: condições
  7. Estruturas de controlo: repetições
  8. Funções e Procedimentos
  9. Estruturas de dados
  10. Sequência de caracteres
  11. Ficheiros e exceções
- 

### Metodologias de ensino (avaliação incluída)

Aulas TP : - Método predominantemente expositivo, com projeção e explicação dos objetivos e conteúdos correspondentes a cada tema. - Prática laboratorial em computador. Resolução e codificação, em linguagem Python, de problemas tipificados, selecionados em conformidade com o conteúdo teórico semanal.

Aulas PL : - Prática laboratorial em computador. Resolução de problemas complementares.

A avaliação compreende duas componentes: um teste ou exame (T/E) e trabalhos práticos (TPs). As duas componentes são avaliadas na escala de 0 a 20. A classificação final é igual a 20% (T/E) + 80% (TPs), com classificação mínima de 7 valores em cada componente. O aluno fica aprovado se obtiver classificação final igual ou superior a 10.

---

### Bibliografia principal

- \* Edelaide Carvalho (2021). Práticas de Python - Algoritmia e Programação. FCA. ISBN: 978-972-722-918-5
- \* Eric Matthes (2016). Curso Intensivo de Python. Uma introdução prática e baseada em projetos à programação. ISBN: 978-85-7522-503-5
- \* Ernesto Costa (2015). Programação em Python - Fundamentos e Resolução de Problemas. FCA. ISBN: 978-972-722-816-4
- \* Luiz Eduardo Borges (2014). Python para Desenvolvedores. Novatec.
- \* Python, <https://www.python.org/>, acedido 2021/06/15

---

**Academic Year** 2022-23

---

**Course unit** PROGRAMMING

---

**Courses** SYSTEMS ENGINEERING AND COMPUTER TECHNOLOGIES (1st cycle)

---

**Faculty / School** INSTITUTE OF ENGINEERING

---

**Main Scientific Area**

---

**Acronym**

---

**CNAEF code (3 digits)** 481

---

**Contribution to Sustainable Development Goals - SGD (Designate up to 3 objectives)** 8; 9

---

**Language of instruction** Portuguese-PT.

---

**Teaching/Learning modality** Presential.

**Coordinating teacher** Mário Alexandre Nobre Saleiro

| Teaching staff                | Type   | Classes  | Hours (*)  |
|-------------------------------|--------|----------|------------|
| Mário Alexandre Nobre Saleiro | PL; TP | TP1; PL1 | 14TP; 42PL |

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

| Contact hours | T | TP | PL | TC | S | E | OT | O | Total |
|---------------|---|----|----|----|---|---|----|---|-------|
|               | 0 | 14 | 42 | 0  | 0 | 0 | 0  | 0 | 130   |

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

Not applied.

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

The subject of Programming aims to provide students with the basics of procedural programming in high-level languages. It is intended that students acquire the concepts essential to algorithmic problem solving, with special emphasis on those that usually arise in computer systems, application structuring, and procedural and data abstraction. The curricular unit consists in solving typified problems and coding in a programming language. Initially, the procedural paradigm is applied, and then object-oriented programming is introduced.

#### Syllabus

- 1.Introduction and Motivation
- 2.Algorithms
- 3.Development environment
- 4.Variables and data types
- 5.Data input and output
- 6.Control structures: conditions
- 7.Control structures: repetitions
- 8.Functions and Procedures
- 9.Data structures
10. Character string
- 11.Files and exceptions

### Teaching methodologies (including evaluation)

#### Theoretical and Practical lessons :

-Method of exposition, explanation and projection of the objectives and contents for each week.

-Practical laboratory computer. Resolution and coding in Python language, typed problems, selected in accordance with the weekly theoretical content.

Practical lessons : Practical laboratory computer.

The evaluation comprises two components: a test or exam (T/E) and practical assignments (TP). The two components are evaluated on the scale of 0 to 20. The final classification is equal to 20% TE + 80% TP, with a minimum classification of 7 values in each component. The student is approved if he or she has a final grade of 10 or higher.

---

### Main Bibliography

Edelaide Carvalho (2021). Práticas de Python - Algoritmia e Programação. FCA. ISBN: 978-972-722-918-5

\* Eric Matthes (2016). Curso Intensivo de Python. Uma introdução prática e baseada em projetos à programação. ISBN: 978-85-7522-503-5

\* Ernesto Costa (2015). Programação em Python - Fundamentos e Resolução de Problemas. FCA. ISBN: 978-972-722-816-4

\* Luiz Eduardo Borges (2014). Python para Desenvolvedores. Novatec.

\* Python, <https://www.python.org/>, acedido 2021/06/15