
1. Designação da unidade curricular

[3697] Informática e Programação / Computer Science and Programming

2. Sigla da área científica em que se insere

INF

3. Duração

Unidade Curricular Semestral

4. Horas de trabalho

108h 00m

5. Horas de contacto

Total: 45h 00m das quais TP: 45h 00m

6. % Horas de contacto a distância

Sem horas de contacto à distância

7. ECTS

4

8. Docente responsável e respetiva carga letiva na Unidade Curricular

[1584] Luciano Alberto do Carmo Jacinto | Horas Previstas: 45 horas

9. Outros docentes e respetivas cargas letivas na unidade curricular

[934] Paulo Jorge Ferreira Arroja Mateus | Horas Previstas: 90 horas

10. Objetivos de aprendizagem e a sua compatibilidade com o método de ensino (conhecimentos, aptidões e competências a desenvolver pelos estudantes).

- A. Dominar uma linguagem de programação estruturada de alto nível.
- B. Desenvolver o raciocínio algorítmico modular na resolução de problemas.
- C. Desenvolver a capacidade de aplicar a abstração procedimental.
- D. Dominar a utilização de ferramentas de computação numérica.
- E. Dominar ferramentas de exploração, organização e representação de dados numa folha de cálculo.

10. Intended Learning objectives and their compatibility with the teaching method (knowledge, skills and competences by the students).

- A. Understanding programming fundamentals.
- B. Ability to analyse a problem and develop an according algorithm.
- C. Ability to use modular algorithmic reasoning in problem solving.
- D. Ability to use numeric computing tools.
- E. Master data exploration, organization and representation tools in a spreadsheet.

11. Conteúdos programáticos

1. Programação: algoritmos e linguagens de programação; desenvolvimento de um programa; tipos de dados elementares; variáveis, operadores e expressões; entrada e saída de dados. Controlo do fluxo: decisão e repetição. Modularidade na programação: funções; variáveis locais; parâmetros. Dados estruturados: armazenamento e manipulação de dados estruturados. Ficheiros: utilização de ficheiros; entrada e saída em ficheiros.
2. O ambiente de computação numérica: escalares, vetores e matrizes. Gráficos. Programação: instruções de controlo e funções.
3. O ambiente folha de cálculo: edição, formatação, validação e proteção de dados. Fórmulas e funções; funções de pesquisa e referência. Bases de dados: ordenar; filtrar e agrupar; tabelas dinâmicas. Gráficos: gráficos dinâmicos.

11. Syllabus

1. Computer programming: algorithms and programming languages; development of a program; elementary data types; variables, operators and expressions; data input and output. Flow control: decision and loops. Modularity: functions; local variables; parameters. Structured data: handling structured data. Files: using files; input and output to files.
2. The numerical computing environment: scalars, vectors and matrices. Graphics. Programming: control statements and functions.
3. The spreadsheet environment: editing, formatting, validation and data protection. Formulas and functions; search and reference functions. Databases: sort; filter and group; pivot tables. Charts: pivot Charts.

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

- O conteúdo programático 1 pretende concretizar os objetivos de aprendizagem A, B e C.
- O conteúdo programático 2 pretende concretizar os objetivos de aprendizagem D.
- O conteúdo programático 3 pretende concretizar os objetivos de aprendizagem E.

12. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes

- The syllabus content 1 intends to achieve the learning outcomes A, B and C.
- The syllabus content 2 intends to achieve the learning outcomes D.
- The syllabus content 3 intends to achieve the learning outcomes E.

13. Metodologias de ensino e de aprendizagem específicas da unidade curricular articuladas com o modelo pedagógico

- Método de ensino:
- Sessões expositivas para apresentação dos conceitos fundamentais de cada um dos conteúdos programáticos 1,2 e 3, sempre acompanhados de exemplos ilustrativos.
- Sessões práticas com a proposta de casos práticos que são resolvidos pelo aluno; estas sessões exigem o envolvimento e uma atitude ativa por parte do aluno na resolução dos problemas propostos.

13. Teaching and learning methodologies specific to the curricular unit articulated with the pedagogical model

Teaching methodology:

Theoretical sessions to present the fundamentals on each content of the syllabus 1, 2 and 3 always through illustrative examples.

Practical sessions with cases to be solved by students; in these sessions students are required to work by themselves in order to achieve the solution to the proposed problems.

14. Avaliação

Método de avaliação: avaliação distribuída com exame final

Os estudantes são avaliados através de dois testes escritos cada um com nota mínima de 8,00 valores e com uma ponderação de 50% na média final. A aprovação é obtida com a média final de 9,50 valores. Se necessário um destes testes escritos pode ser realizado em regime de exame parcial na época normal. A não aprovação em época normal implica a realização de exame na época de recurso.

14. Assessment

Assessment method: Distributed assessment with final exam:

Students are assessed through two written tests each with a minimum grade of 8,00 and weighting 50% in the final average. Approval is obtained with a final average of 9,50. If necessary, one of these written tests can be taken as a partial exam during the normal period. Failure to pass in the normal period implies taking an exam during the appeal period.

15. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular

As sessões expositivas são necessárias para a transmissão dos conceitos fundamentais dos 3 ambientes computacionais estudados 1, 2 e 3.

As sessões práticas são imprescindíveis para atingir os objetivos de aprendizagem A, B, C, D e F dando oportunidade aos alunos de aplicar os conhecimentos adquiridos desenvolvendo a capacidade de analisar, planear e implementar a solução de um problema em qualquer um dos ambientes computacionais estudados 1, 2 e 3.

15. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes

Theoretical sessions are necessary to present the fundamental concepts of each of the studied computational environments 1, 2 and 3.

Practical sessions are essential to achieve the learning outcomes A, B, C, D and E, giving students the opportunity to apply the acquired knowledge, developing the ability to analyse, plan and implement the solution to a problem in any of the studied computational environments.

16. Bibliografia de consulta/existência obrigatória

- AllenDowney, Jeff Elkner, Chris Meyers, Learning with Python: How to Think Like a Computer Scientist, <https://greenteapress.com/wp/learning-with-python/>
- C.H. Swaroop, A Byte of Python v1.92 (for Python 3.0), <https://python.swaroopch.com>
- Coutinho C. (2019). Computação numérica, ISEL, Moodle ISEL
- Coutinho C. (2016). Programação, ISEL, Moodle ISEL
- Coutinho C. (2015). Folha de cálculo, ISEL, Moodle ISEL
- Excel Books and e-Books <https://thesmartmethod.com/>
- Jason Lachniet, 2020, Introduction to GNU Octave, Wytheville Community College, <https://www.wcc.vccs.edu/sites/default/files/Introduction-to-GNU-Octave.pdf>
- Mateus P. (2015). Folha de cálculo - exercícios, ISEL, Moodle ISEL

17. Observações

Unidade Curricular Obrigatória

Data de aprovação em CTC: 2024-07-17

Data de aprovação em CP: 2024-06-26