
1. Designação da unidade curricular

[2523] Programação em Sistemas Computacionais / Computer Systems Programming

2. Sigla da área científica em que se insere

IC

3. Duração

Unidade Curricular Semestral

4. Horas de trabalho

162h 00m

5. Horas de contacto

Total: 67h 30m das quais T: 22h 30m | TP: 45h 00m

6. % Horas de contacto a distância

Sem horas de contacto à distância

7. ECTS

6

8. Docente responsável e respetiva carga letiva na Unidade Curricular

[754] Ezequiel Augusto Cachão Conde | Horas Previstas: 202.5 horas

9. Outros docentes e respetivas cargas letivas na unidade curricular

[818] Mário Henrique Carrasqueira Simões | Horas Previstas: 67.5 horas

[1367] João Pedro Guerreiro da Graça Patriarca | Horas Previstas: 67.5 horas



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

Os estudantes que terminam com sucesso esta unidade curricular serão capazes de:

1. Compreender o modelo de execução de programas escritos em linguagens de alto nível nos sistemas computacionais reais;
2. Conceber e implementar programas de complexidade moderada em linguagem C;
3. Escrever módulos de software em assembly para integrar em programas desenvolvidos em linguagem C;
4. Compreender a arquitetura básica das caches de acesso à memória RAM;
5. Desenvolver software usando o modelo de compilação separada;
6. Compreender as diferenças entre a ligação estática e dinâmica de módulos;
7. Produzir, depurar e testar software modular para ambientes nativos.

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

Students who successfully complete this course unit will be able to:

1. Understand the execution model of programs written in high-level languages in real computer systems;
2. Design and implement programs of moderate complexity in C language;
3. Write software modules in assembly to integrate into programs developed in C language;
4. Understand the basic architecture of RAM caches;
5. Develop software using the separate build model;
6. Understand the differences between the static and dynamic connection of modules;
7. Produce, debug and test modular software for native environments.

11. Conteúdos programáticos

- I. Linguagem C: arrays, estruturas, ponteiros e sua aritmética.
- II. Escrita de programas em assembly. Convenção de chamada a funções: convenções de chamada da linguagem C. Programas envolvendo código em C e assembly. Percurso e manipulação de stack frames.
- III. Hierarquia de memória. Noção de cache. Organização de cache de acesso a RAM e impacto no desempenho dos programas.
- IV. Construção modular de programas: pré-processador; compilação separada; ficheiros cabeçalho e objeto; ligação estática; bibliotecas estáticas.
- V. Sistema de alocação dinâmica de memória sem reciclagem automática.
- VI. Ligação dinâmica de código. Construção e utilização de bibliotecas de ligação



11. Syllabus

- I. Language C: arrays, structures, pointers and their arithmetic.
- II. Write programs in assembly. Function-calling convention: C-language conventions. Programs involving code in C and assembly. Path and manipulation of stack frames.
- III. Memory hierarchy. Cache notion. Organization of RAM access cache and impact on program performance.
- IV. Modular construction of programs: pre-processor; separate compilation; header and object files; static connection; static libraries.
- V. Dynamic memory allocation system without automatic recycling.
- VI. Dynamic code binding. Building and using dynamic link libraries. Dynamic binding at load time and at run time.

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

Nesta unidade curricular, os estudantes compreendem o modelo de execução de programas escritos em linguagens de alto nível no sistema computacional real (pontos II, III, V e VI dos conteúdos programáticos) e adquirem prática de desenvolvimento de aplicações modulares em linguagem C (pontos I e IV). Tais capacidades são fundamentais para a progressão nas áreas de sistemas operativos e de sistemas embebidos. Na componente prática relativa aos pontos IV e VI, os alunos desenvolvem aplicações, em linguagem C, utilizando bibliotecas open-source standard. Finalmente, a introdução às caches de acesso a RAM (ponto III) suporta estudos posteriores na área da programação concorrente.

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

In this curricular unit, students understand the model for executing programs written in high-level languages in the real computing system (points II, III, V and VI of the syllabus) and acquire practice in developing modular applications in C language (points I and IV). Such capabilities are fundamental for progression in the areas of operating systems and embedded systems. In the practical component relating to points IV and VI, students develop applications, in C language, using standard open-source libraries. Finally, the introduction to RAM access caches (point III) supports further studies in the area of concurrent programming.

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

Ensino teórico-prático, estando previstas 30 aulas durante o semestre a que correspondem 67,5 horas de contacto (15 aulas de 3 horas e 15 de 1,5 horas). O tempo total de trabalho do estudante é de 162 horas.

As aulas destinam-se à apresentação dos temas e de exemplos práticos de aplicação. Os tópicos principais são ainda explorados através da realização de trabalhos práticos individuais e em grupo.

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

Theoretical-practical teaching, with 30 classes planned during the semester corresponding to 67.5 contact hours (15 lessons of 3 hours and 15 of 1,5 hours). The total work time of the student is 162 hours. The classes are designed to present topics and practical examples of application. The main topics are further explored through practical group work.

14. Avaliação

Os resultados da aprendizagem são avaliados em avaliação distribuída com exame final (E). A avaliação distribuída é baseada em dois trabalhos práticos. O primeiro trabalho (TP1) versa os pontos I e II do programa e o segundo trabalho prático (TP2) os pontos IV, V e VI. A nota final da UC é obtida pela aplicação da seguinte expressão: $0,1 * TP1 + 0,2 * TP2 + 0,7 * E$. A avaliação dos trabalhos práticos é baseada em relatório e, se necessário, em prova oral de validação da contribuição de cada estudante.

14. Assessment

Learning outcomes are assessed in a distributed assessment with a final exam (E). The distributed assessment is based on two practical assignments. The first assignment (TP1) covers points I and II of the program and the second practical assignment (TP2) covers points IV, V and VI. The final grade for the

UC is obtained by applying the following expression: $0.1 * TP1 + 0.2 * TP2 + 0.7 * E$. The assessment of practical assignments is based on a report and, if necessary, on an oral exam to validate each student's contribution.

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

Os conteúdos programáticos são expostos nas aulas teórico-práticas, complementando uma apresentação interativa das matérias com a realização, pelos estudantes, de pequenos exercícios de consolidação das mesmas. As competências indicadas nos pontos 2, 3, 5, 6 e 7 dos objetivos de aprendizagem são desenvolvidas na realização dos trabalhos de grupo. São dedicadas aulas teórico-práticas para apoio ao desenvolvimento dos trabalhos de grupo, nomeadamente no que se refere à utilização das bibliotecas open-source utilizadas na UC.

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

The program contents are exposed in theoretical-practical classes, complementing an interactive presentation of the subjects with the students carrying out small exercises to consolidate them. The skills indicated in points 2, 3, 5, 6 and 7 of the learning objectives are developed when carrying out group work. Theoretical-practical classes are dedicated to supporting the development of group work, particularly with regard to the use of open-source libraries used in the CU.



Ficha de Unidade Curricular A3ES
Programação em Sistemas Computacionais
Licenciatura em Engenharia Informática e de Computadores
2025-26

16. Bibliografia de consulta/existência obrigatória

B. Kernighan, D. Ritchie, The C Programming Language, 2nd edition, Prentice Hall, 1988. ISBN 9780131103627
R. Bryant, D. O'Hallaron, Computer Systems: A Programmer's Perspective, 3rd edition, Pearson, 2016. ISBN 9780134092669

17. Observações

Unidade Curricular Obrigatória

Data de aprovação em CTC:

Data de aprovação em CP: