

## Ficha de Unidade Curricular – (Versão A3ES 2018-2023)

### 1. Caracterização da Unidade Curricular.

- 1.1. **Designação da unidade curricular** (1.000 carateres).  
Matemática para a Computação Gráfica/ Mathematics for Computer Graphics
- 1.2. **Sigla da área científica em que se insere** (100 carateres).  
MAT
- 1.3. **Duração**<sup>1</sup> (100 carateres).  
Semestral
- 1.4. **Horas de trabalho**<sup>2</sup> (100 carateres).  
162
- 1.5. **Horas de contacto**<sup>3</sup> (100 carateres).  
22.5 T, 22.5 TP, 22.5 PL
- 1.6. **ECTS** (100 carateres).  
6
- 1.7. **Observações**<sup>4</sup> (1.000 carateres).
- 1.7. **Remarks** (1.000 carateres).

### 2. Docente responsável e respetiva carga letiva na Unidade Curricular (preencher o nome completo) (1.000 carateres). Cátia Sofia Peniche Lente Dinis Dias, 135 horas

### 3. Outros docentes e respetivas cargas letivas na unidade curricular (1.000 carateres). João Beleza Teixeira Seixas e Sousa, 67.5 horas

### 4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes). (1.000 carateres).

1. Compreensão das noções básicas da álgebra matricial e geometria afim.
2. Compreensão da renderização gráfica, usando um algoritmo de Ray Tracing.
3. Capacitar o aluno para a implementação de algoritmos para transformar e intersectar representações de objectos geométricos, usando programação orientada por objetos em Python.
4. Capacitar o aluno para a selecção das transformações geométricas adequadas à visualização de uma cena, quando descrita num sistema de coordenadas homogéneas.

### 4. Intended learning outcomes (knowledge, skills and competences to be developed by the students). (1.000 characters).

1. Understanding basic notions from matrix algebra and affine geometry.
2. Understanding graphical rendering using a Ray Tracing algorithm.
3. Enable the student to implement algorithms to transform and intersect representations of geometric objects using object-oriented programming in Python.
4. Enable the student to select geometric transformations suitable for the visualization of a scene, when described in a homogeneous coordinate system.

### 5. Conteúdos programáticos (1.000 carateres).

1. Cálculo Matricial: Álgebra matricial. Sistemas lineares: método de Gauss e classificação. Matriz inversa. Determinante. Valores e vetores próprios. Diagonalização.
2. Cálculo Vetorial: Independência linear, base e dimensão. Norma euclidiana. Ângulo. Produto interno, externo e misto. Base ortonormada em  $\mathbb{R}^n$ .
3. Transformações: Transformações lineares em  $\mathbb{R}^3$ . Transformações de mudança de escala e rotações.

Coordenadas homogêneas. Espaços afins: pontos e direções. Transformações aplicadas a vetores normais.

4. Geometria Tridimensional: Distâncias. Intersecção de um plano com uma reta, intersecção de três planos, transformações no plano. Projeções. Vista de uma cena.
5. Ray Tracing: Intersecção de uma semi-reta com um triângulo. Cálculo dos vetores normais a uma superfície. Vetor de reflexão e de refração a uma superfície.
6. Iluminação: Cor RGB. Fontes de luz. Difusão de luz. Reflexão e refração da luz numa superfície.
7. Programação Orientada por Objetos em Python.

**5. Syllabus (1.000 characters).**

1. Matrix Calculations: Matrix Algebra. Linear systems: Gaussian method and classification. Inverse matrix. Determinant. Eigenvalues and eigenvectors. Diagonalization.
2. Vector Calculus: Linear independence, basis and dimension. Euclidean norm. Angle. Inner, vector and mixed product. Orthonormal basis in  $R^n$ .
3. Transformations: Linear transformations in  $R^3$ . Scale-change transformations and rotations. Homogeneous coordinates. Affine spaces: points and directions. Transformations applied to normal vectors.
4. Three-Dimensional Geometry: Distances. Intersection of a plane with a line, intersection of three planes, transformations in the plane. Projections. View of a scene.
5. Ray Tracing: Intersection of a semi-line with a triangle. Calculation of normal vectors to a surface. Vector of reflection and refraction to a surface.
6. Lighting: RGB color. Sources of light. Diffusion of light. Reflection and refraction of light on a surface.
7. Object Oriented Programming in Python.

**6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular (1.000 caracteres).**

A unidade curricular pretende fundamentar algumas das estruturas matemáticas usadas noutras unidades curriculares. Por forma a facilitar esta ligação e complementar a unidade curricular Matemática Discreta e Programação faz-se aqui também uso da linguagem Python.

Os objetivos de aprendizagem 1 a 4 são abrangidos pelos conteúdos programáticos 1 a 6.

**6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes (1.000 characters).**

The curricular unit aims to provide the mathematical background for other curricular units. The Python language is used here in order to ease the connection between curricular units and to complement the curricular unit Discrete Mathematics and Programming.

The learning outcomes 1 to 4 are covered by sections 1 to 6 of the syllabus.

**7. Metodologias de ensino (avaliação incluída) (1.000 caracteres).**

Ensino teórico-prático, estando previstas aulas com utilização do computador.

Os objetivos de aprendizagem 1 a 4 são avaliados individualmente através de teste global ou exame (NE), de trabalhos práticos realizados ao longo do semestre (NT) e de um projeto (NP). Para obter aprovação é preciso ter nota mínima de 9.5 valores em NE e em  $NF=0.6*NE+0.2*NT+0.2*NP$ .

**7. Teaching methodologies (including assessment) (1.000 characters).**

Theoretical-practical teaching, with some computer using scheduled classes.

The learning objectives 1 to 4 are evaluated individually through global test or exam (NE), practical work carried out throughout the semester (NT) and a project (NP). To obtain approval a minimum grade of 9.5 values in NE and in  $NF = 0.6 * NE + 0.2 * NT + 0.2 * NP$  is required.

**8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular (3.000 caracteres).**

Nas aulas são apresentados os conceitos matemáticos relacionados com os objetivos de aprendizagem 1 a 4, assim como técnicas de resolução de exercícios. São propostos exercícios que permitam consolidar a aprendizagem dos objetivos. As aulas com recurso ao computador, com programação em Python, têm como propósito a consolidação do objetivo 3.

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

*(3.000 characters).*

In classes, the mathematical concepts related to the learning objectives 1 to 4 are presented, as well as techniques for solving problems. Proposed exercises allow the consolidation of the learning objectives. Computer-based classes in Python programming aim to consolidate objective 3.

**9. Bibliografia de consulta/existência obrigatória (1.000 caracteres).**

- E. Lengyel, Mathematics for 3D game programming and computer graphics, 3ed, Cengage Learning, 2011
- G. Farin & D. Hansford, Practical Linear Algebra: A Geometry Toolbox, AK Peters/CRC Press, 2005
- I. Cabral & C. Perdigão & C. Saiago, Álgebra Linear: teoria, exercícios resolvidos e exercícios propostos com soluções, 5ed., Escolar Editora, 2018
- J. F. Hughes & A. van Dam & M. McGuire & D. F. Sklar & J. D. Foley & S. K. Feiner & K. Akeley, Computer Graphics: Principles and Practice, 3ed, Addison-Wesley Professional, 2014.
- A. Howard & C. Rorres, Álgebra linear com aplicações, 10ed., Bookman, 2012

---

<sup>1</sup> Anual, semestral, trimestral, ...

<sup>2</sup> Número total de horas de trabalho.

<sup>3</sup> Discriminadas por tipo de metodologia adotado (T - Ensino teórico; TP - Ensino teórico-prático; PL - Ensino prático e laboratorial; TC - Trabalho de campo; S - Seminário; E - Estágio; OT - Orientação tutorial; O - Outro).

<sup>4</sup> Assinalar sempre que a unidade curricular seja optativa.