
1. Caracterização da Unidade Curricular

1.1 Designação

[4227] Ambientes Virtuais Interativos e Inteligentes / Interactive and Intelligent Virtual Environments

1.2 Sigla da área científica em que se insere

INF

1.3 Duração

Unidade Curricular Semestral

1.4 Horas de trabalho

162h 00m

1.5 Horas de contacto

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

1.6 ECTS

6

1.7 Observações

Unidade Curricular Obrigatória

2. Docente responsável

[760] Arnaldo Joaquim Castro Abrantes

3. Docentes e respetivas cargas letivas na unidade curricular

Não existem docentes definidos para esta unidade curricular

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

No final desta unidade curricular os estudantes serão capazes de:

1. Compreender o papel da Geração Procedimental de Conteúdos (GPC) e da Inteligência Artificial (IA) na criação de ambientes virtuais interativos para simulações/jogos, capazes de emular o mundo real, nas suas componentes sensoriais (colisões, sons, imagens), cognitivas (inteligência, adaptação, aprendizagem) e sociais (comunicação, cooperação, competição);
2. Conhecer e aplicar técnicas de GPC, nomeadamente para criação de terrenos e níveis (relevo, vegetação, rios, estradas, cidades, biomas);
3. Conhecer e aplicar técnicas de IA para treinar agentes autónomos (NPCs) a desempenhar uma determinada tarefa, individual ou coletiva;
4. Utilizar simulações/jogos como ambientes virtuais para treinar agentes autónomos (e.g., condução de veículo), com o objetivo de posterior aplicação em ambientes reais (robótica);
5. Saber como aplicar a interação com o utilizador/jogador para guiar os processos de GPC e do treino dos agentes.

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

At the end of this course students will be able to:

1. Understand the role of Procedural Content Generation (PCG) and Artificial Intelligence (AI) in the creation of interactive virtual environments for simulations/games capable of emulating the real world in its sensory components (collisions, sounds, images), cognitive (intelligence, adaptation, learning) and social (communication, cooperation, competition);
2. Know and apply PCG techniques, namely for creating terrains and levels (relief, vegetation, rivers, roads, cities, biomes);
3. Know and apply IA techniques to train autonomous agents (NPC) to perform a certain task, individual or collective;
4. Use simulations/games as virtual environments to train autonomous agents (e.g., driving a vehicle), with the objective of later application in real environments (robotics);
5. Know how to apply the interaction with the user/player to guide the processes of PCG and the training of the agents (human-in-the-loop).

5. Conteúdos programáticos

1. Descrição geral da plataforma Unity (editor e motor de jogo);
2. Criação de mundos virtuais, de forma procedimental, utilizando blocos (tipo Minecraft);
3. Utilização de técnicas de GPC, tais como ruído de Perlin, autómatos celulares, gramáticas de Lindenmayer (e.g., para criação de cavernas, rios, estradas);
4. Perspetiva geral de metodologias de aprendizagem automática para treino de agentes autónomos: algoritmos genéticos, redes neuronais, aprendizagem por imitação, aprendizagem por reforço;
5. Utilização da arquitetura do Unity ML-Agents Toolkit para treinamento de agentes autónomos, baseada em 3 tipos de objetos, organizados de forma hierárquica: Agente, Cérebro e Academia;
6. Utilização do pacote TensorFlow e API python para treinamento offline e posterior exportação dos resultados para o Unity;
7. Desenvolvimento de ambientes de simulação/jogos e respetivo treino de agentes (e.g., navegação em labirintos, condução autónoma de veículos, mini jogo de futebol, ecossistemas).

5. Syllabus

1. Overview of the Unity platform (game editor and engine);
2. Creation of virtual worlds, procedurally, using blocks (type Minecraft);
3. Use of PCG techniques, such as Perlin's noise, cellular automata, Lindenmayer's grammars (e.g., for creating caves, rivers, roads);
4. General perspective of machine learning methodologies for training autonomous agents: genetic algorithms, neural networks, imitation learning, reinforcement learning;
5. Use of the architecture of the Unity ML-Agents Toolkit for training autonomous agents, which is based on 3 types of objects, organized hierarchically: Agent, Brain and Academy;
6. Use of the TensorFlow package and its python API for offline training (and then exporting the results to Unity);
7. Development of simulation/game environments and respective training of agents (e.g., maze navigation, autonomous vehicle driving, mini-soccer game, ecosystems).

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

A unidade curricular tem como objetivo fornecer os conceitos necessários à criação de ambientes virtuais interativos de forma programática, recorrendo a técnicas de Geração Procedimental de Conteúdos (GPC) e de Inteligência Artificial (IA).

A UC apresenta técnicas comuns em GPC, nomeadamente ruído de Perlin, autómatos celulares e gramáticas de Lindenmayer, para a criação de cenários estáticos e dinâmicos.

O comportamento desejado dos personagens que habitam esses cenários é obtido recorrendo a técnicas de aprendizagem automática, tais como algoritmos genéticos, redes neuronais, aprendizagem por imitação e por reforço.

É usada uma ferramenta específica, disponibilizada pelo Unity, para treinamento de agentes autónomos, bem como pacotes externos (e.g., TensorFlow) para treinamento offline (exportação posterior para o Unity). O aluno concretiza os conhecimentos adquiridos através da elaboração de trabalhos e projetos, envolvendo o desenvolvimento de pequenas simulações/jogos.

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

The curricular unit aims to provide the concepts necessary to create interactive virtual environments in a programmatic way, using Procedural Content Generation (PCG) and Artificial Intelligence (AI) techniques.

The UC presents common techniques in PCG, namely Perlin noise, cellular automata and Lindenmayer grammars, for the creation of static and dynamic scenarios. The desired behaviour of the characters that inhabit these scenarios is obtained by using machine learning techniques, such as genetic algorithms, neural networks, imitation learning and reinforcement learning.

The specific Unity ML-Agents Toolkit is used to train autonomous agents, complemented with external packages (e.g., TensorFlow) used for offline training (exporting results to Unity). The student concretizes the acquired knowledge through the elaboration of works and projects, involving the development of small simulations/games.

7. Metodologias de ensino (avaliação incluída)

A carga horária está dividida em 22,5 horas teórico-práticas (15 aulas de 1H30M) e 45 horas de prática laboratorial (15 aulas de 3 horas). As aulas destinam-se à apresentação dos temas e à exemplificação de casos práticos de utilização. É utilizada uma metodologia de ensino suportada no desenvolvimento de pequenos projetos realizados ao longo do semestre, utilizando a plataforma Unity. A avaliação é distribuída com exame final.

A classificação final será obtida através da média ponderada dos seguintes elementos de avaliação: exame (30%), média aritmética das notas dos trabalhos realizados ao longo do semestre (30%) e projeto final com discussão (40%). Para obter aprovação, o aluno terá que obter média final igual ou superior a 9.5 (escala 0-20) com nota mínima no exame de 9.5 e nota mínima na componente prática (trabalhos e projeto) também de 9.5. Todos os elementos de avaliação são pedagogicamente fundamentais.

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

The workload is divided into 22.5 theoretical-practical hours (15 lessons of 1 hour and 30 minutes) and 45 hours of laboratory practice (15 lessons of 3 hours). The lectures are aimed at presenting the topics and giving examples of practical use cases. The teaching methodology is based on the development of small projects throughout the semester, using the Unity platform. Assessment is distributed with a final exam.

The final grade will be obtained from the weighted average of the following assessment elements: exam (30%), arithmetic average of the grades of the work carried out during the semester (30%) and final project with discussion (40%). To be approved, the student must obtain a final average of 9.5 or higher (0-20 scale) with a minimum mark in the exam of 9.5 and a minimum mark in the practical component (work and project) of 9.5. All assessment components are pedagogically fundamental.

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

Os temas correspondentes aos resultados de aprendizagem, na sua componente conceptual, são estudados em aulas teóricas específicas e concretizados em casos práticos e projetos desenvolvidos ao longo do semestre, em aulas teórico-práticas e de modo autónomo pelos alunos.

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

The themes corresponding to the learning outcomes, in their conceptual component, are studied in specific theoretical classes. These topics are then concretized in practical cases and projects, developed during the semester, autonomously by the students, in theoretical-practical classes.

**9. Bibliografia de
consulta/existência obrigatória**

Yannakakis, G. (2018), Artificial Intelligence and Games, Springer
Short, T. (2017), Procedural Generation in Game Design, A K Peters/CRC Press
Dutta, S. (2018), Reinforcement Learning with TensorFlow: A beginner's guide to designing self-learning systems with TensorFlow and OpenAI Gym, Packt Publishing
Palacios, J. (2018), Unity 2018 Artificial Intelligence Cookbook: Over 90 recipes to build and customize AI entities for your games with Unity, Packt Publishing
Lanham, M. (2018), Learn Unity ML-Agents - Fundamentals of Unity Machine Learning: Incorporate new powerful ML algorithms such as Deep Reinforcement Learning for games, Packt Publishing

10. Data de aprovação em CTC 2024-07-17

11. Data de aprovação em CP 2024-06-26