

# Projecto prático de Programação Orientada aos Objectos

MiEI + LCC

edição 20/21

## Contents

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Requisitos</b>	<b>2</b>
2.1	Requisitos base de gestão das entidades . . . . .	2
2.2	Calcular o resultado de um jogo . . . . .	2
2.3	Efectuar a simulação de um jogo . . . . .	2
<b>3</b>	<b>Relatório</b>	<b>2</b>
<b>4</b>	<b>Salvaguarda do estado da aplicação</b>	<b>3</b>
<b>5</b>	<b>Patamares de classificação do projecto</b>	<b>3</b>
<b>6</b>	<b>Cronograma</b>	<b>3</b>

## Abstract

Este documento apresenta o enunciado do projecto prático de Programação Orientada aos Objectos. Numa primeira fase identificam-se os requisitos principais e posteriormente será fornecida mais informação.

## 1 Introdução

Considere que se pretende construir um programa que seja semelhante com o conhecido jogo Football Manager. Pretende criar um sistema de gestão e simulação de equipas de um determinado desporto (p.e. futebol). Cada equipa é constituída por um conjunto fixo de jogadores, sendo que em cada jogo podem jogar  $N$  jogadores titulares e utilizarem-se  $M$  suplentes. Cada jogador possui um conjunto de características e, em determinado momento, possui um valor de habilidade (que quanto mais elevado melhor é o desempenho esperado do jogador).

As características variam de desporto para desporto, mas consideremos que o principal requisito deste programa é suportar equipas de futebol. As características base de um jogador de futebol são: *velocidade*, *resistência*, *destreza*, *impulsão*, *jogo de cabeça*, *remate* e *capacidade de passe*.

Existem vários tipos de jogadores, a saber:

1. guarda-redes
2. defesas
3. médios
4. avançados

## 5. laterais

Os guarda-redes acrescentam ao conjunto de características a *elasticidade*, que é apenas associada a este tipo de jogadores. Cada tipo de jogador possui uma fórmula que atribui pesos diferentes às diversas características e que determina um valor global de habilidade. A habilidade global de uma equipa é função da habilidade de cada um dos jogadores. Um jogo entre duas equipas tem um resultado probabilístico, mas a equipa com um valor de habilidade superior tem mais probabilidade de ganhar.

## 2 Requisitos

O trabalho proposto tem vários níveis de requisitos, desde os mais básicos até aos mais complexos. Para que os vários níveis de requisitos identificados sejam considerados cumpridos o programa deve respeitar os princípios da programação por objectos, ser robusto no tratamento de erros bem como gravar, e também carregar, o estado do programa para um ficheiro de objectos. Também deverá suportar o carregamento de um estado em modo texto, que a equipa docente posteriormente fornecerá.

### 2.1 Requisitos base de gestão das entidades

Como requisitos base pretende-se ter a capacidade de criar jogadores e equipas e associar os vários jogadores a uma equipa. O programa deverá permitir consultar os jogadores e as equipas e calcular os valores de habilidade para um jogador ou para uma equipa. Deverá ser também possível mudar um jogador de equipa, sendo necessário registar o historial de um determinado jogador (as equipas por onde passou).

Nota máxima: 12 valores

### 2.2 Calcular o resultado de um jogo

O programa deverá possibilitar a realização de um jogo entre duas equipas. Deverá ser possível definir as equipas titulares e as substituições a efectuar e em função das equipas calcular o vencedor do jogo.

Nota máxima: 16 valores

### 2.3 Efectuar a simulação de um jogo

O programa deverá conseguir simular o jogo, propondo a divisão do tempo do jogo em várias partes e em cada uma delas calcular o que acontece em termos de jogo. Por exemplo, pode arbitrar-se que em cada uma das divisões do tempo uma das equipas é responsável por atacar e em função disso a jogada pode resultar em golo ou então em perda de bola para o adversário. No final do tempo o programa deverá dizer quem ganhou e qual o resultado.

Nota máxima: 20 valores

## 3 Relatório

O relatório deve descrever o trabalho realizado para desenvolver a aplicação solicitada. No mínimo, devem ser abordados os seguintes pontos:

- Capa com identificação da Unidade Curricular e do grupo (nome, número e fotografia).
- Descrição da arquitectura de classes utilizada (classes, atributos, etc.) e das decisões que foram tomadas na sua definição. Deverá ser entregue um **Diagrama de Classes** com a arquitectura de classes que suporta o programa desenvolvido.
- Descrição da aplicação desenvolvida (ilustração das funcionalidades).

## 4 Salvaguarda do estado da aplicação

O programa deve permitir que em qualquer momento se possa guardar em ficheiro a informação existente em memória sobre a informação relevante das entidades. A gravação deve ser feita de forma a permitir que o estado que foi gravado seja recuperado novamente. Na altura da entrega do projecto deve ser também entregue um estado (guardado em ficheiro) que possa ser carregado durante a apresentação.

## 5 Patamares de classificação do projecto

Como foi referido atrás este projecto tem previstos três patamares de dificuldade em função dos requisitos anteriormente identificados.

1. Requisitos base de gestão das entidades: nota máxima 12 valores;
2. Item anterior mais cálculo do resultado de um jogo: nota máxima 16 valores;
3. Itens anteriores mais realização de simulação de um jogo: nota máxima 20 valores.

Para que um projecto possa ser considerado positivo (com nota maior ou igual a 10) será necessário que se consiga efectuar, durante a apresentação, o registo e posterior consulta da informação de jogadores e de equipas e determinar o nível de habilidade de uma equipa.

Obviamente que a nota a atribuir irá reflectir a estruturação da solução que deverá respeitar as normas da programação orientada aos objectos, cf aulas teóricas, nomeadamente o encapsulamento, a abstracção de implementação e a capacidade de a aplicação evoluir de forma controlada (eg: através da incorporação de novos tipos de desportos e de equipas e de tipos de jogadores).

## 6 Cronograma

A entrega do projecto far-se-á de forma faseada, nas seguintes datas:

1. Entrega de projecto com uma versão inicial das declarações das classes (pelo menos com as variáveis de instância) e um diagrama de classes. Entrega da composição do grupo.  
**Data Limite:** *1 de Maio* (esta fase é eliminatória, isto é, os grupos que não entregarem não poderão submeter o projecto final).
2. Entrega final de código, ficheiro com os dados a carregar durante a apresentação e relatório de projecto (feita por via electrónica no elearning)  
**Data Limite:** *12 de junho de 2021 - data acordada com a direcção de curso*
3. Apresentação do projecto  
**Semana de:** *21 a 26 de junho - data acordada com a direcção de curso*