

Projeto POO

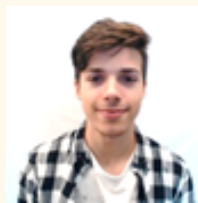
Football Manager

Grupo 74



Source: <https://www.footballmanager.com/#desktop>

Autores do Trabalho



Vasco Oliveira, nº93208



Diogo Matos, nº93234



Miguel Fernandes, nº94269

Introdução

Este trabalho foi desenvolvido no domínio da cadeira de programação orientada a objetos e pretende simular ao jogo Football Manager. Para tal foi utilizada a linguagem de programação Java e os conhecimentos adquiridos ao longo do semestre nesta disciplina.

Estrutura geral da aplicação

Para desenvolver este projeto decidimos utilizar a estratégia MVC(Model, View, Controller) aprendida nas aulas. Assim, apresenta-se um diagrama da estrutura utilizada:

Model:

Jogo

Jogo	
equipaCasa	SimpleEntry<String, Equipa>
equipaFora	SimpleEntry<String, Equipa>
golosCasa	int
golosFora	int
date	LocalDate
jogadoresCasa	Map<Integer, Jogador>
jogadoresFora	Map<Integer, Jogador>
substituicoesCasa	Map<Integer, Integer>
substituicoesFora	Map<Integer, Integer>
Jogo()	
Jogo(Equipa, Equipa, List<Integer>, List<Integer>, Map<Integer, Integer>, Map<Integer, Integer>)	
Jogo(Jogo)	
Jogo(String, String, int, int, LocalDate, List<Integer>, Map<Integer, Integer>, List<Integer>, Map<Integer, Integer>)	
Jogo(String, String, int, int, LocalDate, List<Integer>, Map<Integer, Integer>, List<Integer>, Map<Integer, Integer>, Equipa, Equipa)	
clone()	Jogo
difsEquipas()	double
equals(Object)	boolean
equipaTitular(List<Jogador>)	Map<Integer, Jogador>
escolheGuardaRedes(Map<String, List<Jogador>>)	Jogador
escolhePosicao(Map<String, List<Jogador>>, String, int, String, String)	List<Jogador>
getData()	LocalDate
getEquipaCasa()	SimpleEntry<String, Equipa>
getEquipaFora()	SimpleEntry<String, Equipa>
getGolosCasa()	int
getGolosFora()	int
getJogadoresCasa()	Map<Integer, Jogador>
getJogadoresFora()	Map<Integer, Jogador>
getSubstituicoesCasa()	Map<Integer, Integer>
getSubstituicoesFora()	Map<Integer, Integer>
melhoresPosicao(List<Jogador>, String)	List<Jogador>
melhoresTodasPosicoes(List<Jogador>)	Map<String, List<Jogador>>
overallPosicao(String, int)	double
overallTitulares(List<Jogador>)	double
parse(String)	Jogo
probs(double, double)	double
setDate(LocalDate)	void
setEquipaCasa(String, Equipa)	void
setEquipaFora(String, Equipa)	void
setGolosCasa(int)	void
setGolosFora(int)	void
setJogadoresCasa(Equipa)	void
setJogadoresFora(Equipa)	void
simulecaJogo()	void
situacoesGolos(double, double)	void
substituicoes()	void
substituicoes2()	void
toString()	String
toString2()	String

Jogador

Jogador	
nome	String
numero	int
velocidade	int
resistencia	int
destreza	int
impulsao	int
jogoDeCabeca	int
remate	int
capacidadeDePasse	int
histEquipas	List<String>
Jogador()	
Jogador(Jogador)	
Jogador(String, int, int, int, int, int, int, int)	
addEquipa(String)	void
clone()	Jogador
equals(Object)	boolean
getCapacidadeDePasse()	int
getDestreza()	int
getHistEquipas()	List<String>
getImpulsao()	int
getJogoDeCabeca()	int
getNomeJogador()	String
getNumeroJogador()	int
getRemate()	int
getResistencia()	int
getVelocidade()	int
overall()	double
setCapacidadeDePasse(int)	void
setDestreza(int)	void
setHistEquipas(List<String>)	void
setImpulsao(int)	void
setJogoDeCabeca(int)	void
setNomeJogador(String)	void
setNumeroJogador(int)	void
setRemate(int)	void
setResistencia(int)	void
setVelocidade(int)	void
toString()	String

Equipa

Equipa	
clube	String
jogadores	Map<Integer, Jogador>
Equipa()	
Equipa(Equipa)	
Equipa(String)	
Equipa(String, ArrayList<Jogador>)	
Equipa(String, int, int, ArrayList<Jogador>)	
addJogador(Jogador)	void
clone()	Equipa
equals(Object)	boolean
existeJogador(int)	Jogador
getClube()	String
getJogadores()	Map<Integer, Jogador>
overallEquipa()	double
parse(String)	Equipa
removeJogador(Jogador)	void
setClube(String)	void
setJogadores(ArrayList<Jogador>)	void
toString()	String

Powered by yfiles

LigaPOO

LigaPOO	
jogSemEquipa	Map<Integer, Jogador>
equipas	Map<String, Equipa>
jogos	List<Jogo>
LigaPOO()	
LigaPOO(LigaPOO)	
LigaPOO(Map<String, Equipa>, List<Jogo>, Map<Integer, Jogador>)	
adicionaEquipa(Equipa)	void
adicionaJogSemEquipa(Jogador)	void
clone()	LigaPOO
existeEquipa(String)	Equipa
getEquipas()	Map<String, Equipa>
getJogSemEquipa()	Map<Integer, Jogador>
getJogos()	List<Jogo>
jogosDoDia(LocalDate)	List<Jogo>
procuraJogador(String, int)	Jogador
readFromBinary(String)	LigaPOO
removeJogSemEquipa(int)	void
saveToBinary(String)	void
setEquipas(Map<String, Equipa>)	void
setJogSemEquipa(Map<Integer, Jogador>)	void
setJogos(List<Jogo>)	void
simulaJogo(Jogo)	Jogo
simulaJogo(String, String, List<Integer>, List<Integer>, Map<Integer, Integer>, Map<Integer, Integer>)	Jogo
transferencia(String, String, int)	void
transferencia(String, int)	void

Medio

Medio	
recuperarBolas	int
Medio()	
Medio(Medio)	
Medio(String, int, int, int, int, int, int, int, int)	
Medio(String, int, int, int, int, int, int, int, int)	
clone()	Medio
equals(Object)	boolean
getRecuperarBolas()	int
overall()	double
overallCentral()	double
parse(String)	Medio
setRecuperarBolas(int)	void
toString()	String

Guarda Redes

GuardaRedes	
elasticidade	int
GuardaRedes()	
GuardaRedes(GuardaRedes)	
GuardaRedes(String, int, int, int, int, int, int, int, int)	
GuardaRedes(String, int, int, int, int, int, int, int, int)	
clone()	GuardaRedes
equals(Object)	boolean
getElasticidade()	int
overall()	double
parse(String)	GuardaRedes
setElasticidade(int)	void
toString()	String

Lateral

Lateral	
cruzamento	int
Lateral()	
Lateral(Lateral)	
Lateral(String, int, int, int, int, int, int, int, int)	
clone()	Lateral
equals(Object)	boolean
getCruzamento()	int
overall()	double
overallDefensivo()	double
overallOfensivo()	double
parse(String)	Lateral
setCruzamento(int)	void
toString()	String

Avançado

Avançado	
Avançado()	
Avançado(Avançado)	
Avançado(String, int, int, int, int, int, int, int, int)	
clone()	Avançado
equals(Object)	boolean
overall()	double
overallCentral()	double
parse(String)	Avançado
toString()	String

Defesa

Defesa	
Defesa()	
Defesa(Defesa)	
Defesa(String, int, int, int, int, int, int, int, int)	
clone()	Defesa
equals(Object)	boolean
overall()	double
overallCentral()	double
parse(String)	Defesa
toString()	String

Defensivo

Defensivo	
overallDefensivo()	double

Ofensivo

Ofensivo	
overallOfensivo()	double

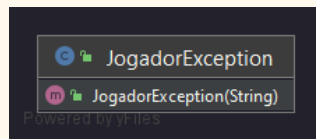
Parser

Parser	
Parser()	
lerFicheiro(String)	List<String>
parse(Map<String, Equipa>, Map<Integer, Jogador>, List<Jogo>)	void

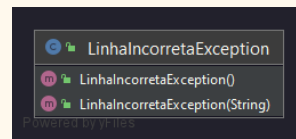
Jogador Comparator

JogadorComparator	
JogadorComparator()	
compare(Jogador, Jogador)	int

Jogador Exception

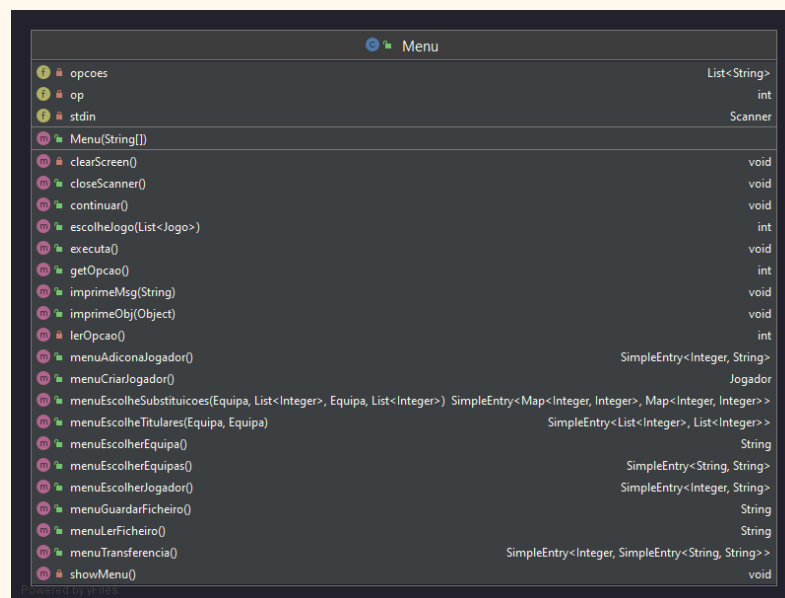


Linha Incorreta Exception



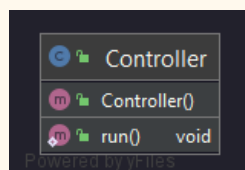
View:

Menu

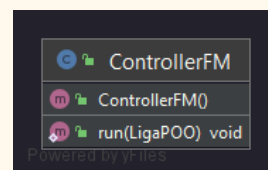


Controller:

Controller

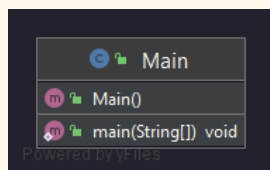


ControllerFM



Fora do contexto MVC

Main



Escolhas feitas e seus porquês

Optamos por este estruturamento uma vez que nos permite manipular livremente as classes que pretendemos. O facto da classe Jogador ser superclasse dos diferentes tipos de jogadores(Avançado, guarda redes, defesa...) facilita exponencialmente o acesso às diferentes estruturas e variáveis que se tornariam muito repetitivas caso contrário.

Para além disso, optamos também por tornar a classe jogador abstrata, uma vez que vários métodos iriam ter necessidade de ser definidos, individualmente, e de maneira diferente, nos diferentes tipos de jogadores, o que causaria problemas caso contrário.

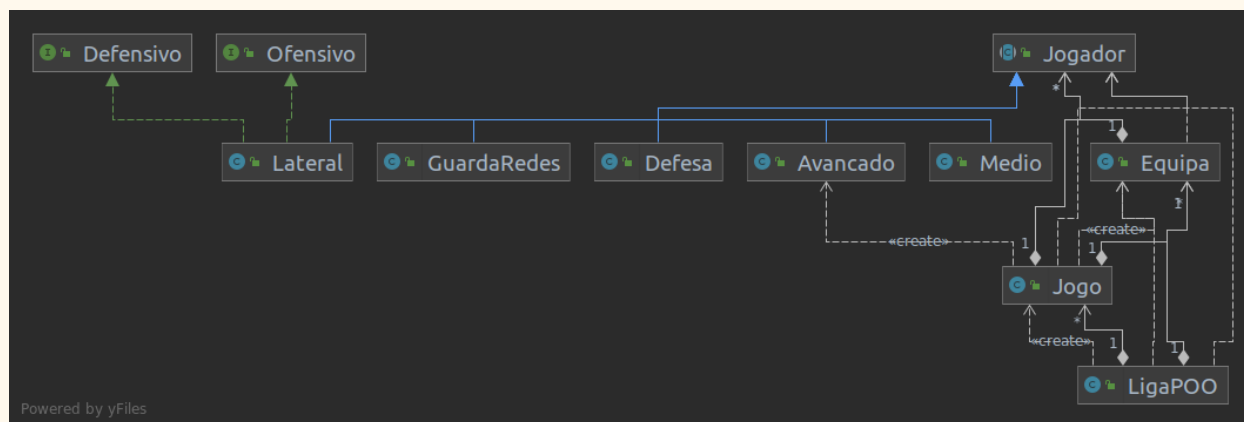
Podemos também reparar que existem duas interfaces, defensivo e ofensivo, que não foram definidas como classes uma vez que não constam ser tipos de jogadores em si, mas apenas uma propriedade que alguns jogadores têm.

Na classe equipa, fora decidido ter um mapa de jogadores, aos quais a chave correspondia o seu número individual. Este método facilita preferencialmente a procura.

Além disso, criamos a classe LigaPOO, que contém diferentes funções importantes para a simulação de jogo e tratamento de dados.

Finalmente, o resto das estruturas foram baseadas no método MVC , já anteriormente referido.

Diagrama com dependências



Requisitos

Quanto aos 3 requisitos pedidos no enunciado o grupo dedicou-se arduamente em tentar cumprir todos, porém não foi possível.

- Requisitos base de gestão das entidades

Nesta alínea todas as funcionalidades foram cumpridas e estão todas operacionais.

- Calcular o resultado de um jogo

No nosso programa é possível calcular o resultado do Jogo onde o resultado tem por base a média dos Jogadores em campo que influencia o número de ataques de cada equipa e depois existe uma probabilidade de marcar golo que resulta de uma comparação da qualidade ofensiva da equipa que ataca sobre a qualidade defensiva e do guarda-redes da equipa que defende.

- Efectuar a simulação de um jogo

Optamos por seguir o conselho do enunciado e dividir o jogo em instantes de tempo, neste caso 9. Como referi anteriormente a probabilidade de cada equipa realizar o ataque depende da qualidade de todo o coletivo. Para marcar golo, isto já só depende dos fatores ofensivos e defensivos das respetivas equipas. As 3 substituições são realizadas no instante 6 do jogo.

Quanto aos objetivos 7.0 e 7.1 estes foram inicializados porém não conseguiram ser concluídos. A interface ofensivo e defensivo ia permitir que jogadores jogassem fora das suas posições e ainda desenvolvemos várias funções na classe Jogo onde permitiria escolher o melhor 11 inicial e realizar as substituições de uma forma inteligente, isto é, tirar os jogadores com menor resistência substituindo pelos suplentes da mesma posição com o melhor média de habilidades.

A estratégia utilizada para definir o melhor 11 inicial era dividir os jogadores por posições ordenadas pelo overall, depois consoante a tática escolhida ou então a tática que favorecesse mais equipa eram escolhidos os jogadores para cada posição. Para cada posição eram dadas 3 hipóteses, sendo a primeira colocar sempre o jogador na sua posição de raiz. Por exemplo para escolher o avançado, para uma tática de 4-3-3: da lista ordenada tentava-se tirar os 3 melhores, se não existissem 3, a segunda opção era a lista de médio, pois normalmente jogam numa posição avançada e conseguem muitas vezes desempenhar o papel de avançado e a terceira opção era escolher da lista de Defesa uma vez que como diz no enunciado este costumam ter bom jogo de cabeça.

Esta estratégia foi implementada porém não foi possível deixá-la operacional.

Funcionalidade da aplicação

Quando se inicia a aplicação é possível carregar dados de um jogo realizado anteriormente, ou então inicializar um jogo novo apenas com os dados fornecidos no “logs.txt”.

Dentro do jogo é possível realizar 10 ações diferentes, sendo uma destas guardar o estado do jogo.

Simular jogo

O simular jogo é como se fosse um jogo para 2 utilizadores, cada jogador escolhe a equipa com quer jogar, a equipa titular e as substituições que deseja fazer. A partir dos dados escolhidos é simulado um jogo e apresenta aos utilizadores o resultado.


```

Opção: 1
Jogador 1:
Nome da equipa:
Mahler Athletic
Jogador 2
Nome da equipa:
Sporting Club Chaplin
Clube: Mahler Athletic (Overall: 59,34)
Avancado: 32 -> Jose Pedro Pinto Teixeira da Cunha; 89; 63; 59; 66; 60; 50; 25; (Overall:61,05); Antigos clubes:[]
Avancado: 2 -> Bruno Filipe Jardim Machado; 90; 73; 56; 60; 41; 72; 53; (Overall:64,90); Antigos clubes:[]
Avancado: 39 -> Guilherme Gil Rocha Goncalves; 68; 31; 13; 65; 30; 61; 71; (Overall:47,45); Antigos clubes:[]
Avancado: 22 -> Pedro Antonio Pires Correia Leite Sequeira; 55; 54; 8; 93; 94; 70; 68; (Overall:66,15); Antigos clubes:[]
Avancado: 28 -> Raul Arieira Parente; 64; 54; 87; 21; 83; 82; 72; (Overall:69,20); Antigos clubes:[]
Defesa: 8 -> Fabio Fernandes Silva; 51; 91; 64; 58; 14; 75; 42; (Overall:54,25); Antigos clubes:[]
Defesa: 48 -> Nelson Ramos Alves; 59; 56; 51; 52; 67; 59; 42; (Overall:53,90); Antigos clubes:[]
Defesa: 27 -> Pedro Miguel Sampaio Faria; 58; 72; 52; 51; 61; 57; 51; (Overall:55,90); Antigos clubes:[]
Guarda redes: 36 -> Eduardo Costa de Magalhaes; 58; 45; 94; 36; 6; 59; 47; (Overall:64,64); Antigos clubes:[]
Lateral: 40 -> Barbara Paulista de Faria; 58; 24; 70; 53; 54; 89; 65; (Overall:46,78); Antigos clubes:[]
Lateral: 15 -> Diogo da Costa e Silva Lima Rebelo; 57; 66; 70; 96; 60; 74; 68; (Overall:57,60); Antigos clubes:[]
Lateral: 20 -> Mariana Dinis Rodrigues; 53; 62; 56; 64; 0; 92; 59; (Overall:50,90); Antigos clubes:[]
Lateral: 25 -> Tomas Cardoso Francisco; 71; 58; 70; 0; 71; 73; 84; (Overall:65,77); Antigos clubes:[]
Lateral: 29 -> Tiago Rafael Fernandes Guimaraes; 61; 58; 58; 61; 37; 95; 55; (Overall:67,25); Antigos clubes:[]
Lateral: 30 -> Pedro Nuno Fernandes Freitas; 53; 71; 63; 45; 5; 63; 66; (Overall:63,70); Antigos clubes:[]
Medio: 11 -> Carlos Daniel da Silva Santos; 56; 75; 84; 63; 65; 56; 69; (Overall:67,00); Antigos clubes:[]
Medio: 12 -> Joao Guilherme Rodrigues Reis; 67; 73; 61; 72; 73; 54; 43; (Overall:51,05); Antigos clubes:[]
Medio: 17 -> Francisco Jose Pereira Teofilo; 68; 66; 96; 74; 69; 57; 76; (Overall:71,25); Antigos clubes:[]
Medio: 21 -> Goncalo Miguel Leao Barros Oliveira Pinto; 59; 3; 1; 88; 52; 54; 72; (Overall:48,70); Antigos clubes:[]
Medio: 23 -> Joao Pedro Martins da Silva Freitas Cardoso; 3; 64; 69; 54; 59; 49; 90; (Overall:59,45); Antigos clubes:[]

Jogador 1 - Escolha o seu onze inicial(Por numero separado por -enter-)
34
35
36
37
38
39
40
41
42
43
44
45
46

```

```

Clube: Sporting Club Chopin (Overall: 59,34)
Avancado: 35 -> Meriem Khammassi; 93; 51; 44; 58; 56; 98; 63; (Overall:70,65); Antigos clubes:[]
Avancado: 37 -> Daniel Torres Azevedo; 46; 63; 63; 55; 73; 60; 72; (Overall:61,35); Antigos clubes:[]
Avancado: 23 -> Beatriz Fernandes Oliveira; 69; 32; 53; 72; 18; 67; 54; (Overall:50,70); Antigos clubes:[]
Defesa: 7 -> Pedro Paulo Costa Pereira; 52; 26; 71; 59; 70; 53; 96; (Overall:65,50); Antigos clubes:[]
Defesa: 8 -> Joao Pedro Espilma Goulart; 29; 74; 36; 62; 66; 78; 52; (Overall:52,00); Antigos clubes:[]
Defesa: 40 -> Goncalo Couto dos Santos; 53; 26; 27; 84; 59; 93; 96; (Overall:56,50); Antigos clubes:[]
Defesa: 44 -> Joao Manuel Novais da Silva; 51; 77; 59; 60; 10; 65; 60; (Overall:53,25); Antigos clubes:[]
Defesa: 48 -> Andre Goncalves Pinto; 54; 55; 37; 62; 22; 51; 92; (Overall:50,95); Antigos clubes:[]
Guarda redes: 47 -> Catarina Morales de Los Rios Martins; 11; 82; 73; 72; 73; 75; 65; (Overall:80,43); Antigos clubes:[]
Guarda redes: 20 -> Goncalo Leal Da Mota Meireles Moreira; 78; 55; 69; 39; 90; 52; 73; (Overall:68,53); Antigos clubes:[]
Guarda redes: 22 -> Diogo Manuel Brito Pires; 73; 61; 18; 50; 75; 71; 18; (Overall:21,46); Antigos clubes:[]
Lateral: 38 -> Joao Antonio Ribeiro Alves; 82; 50; 49; 53; 75; 75; 17; (Overall:48,30); Antigos clubes:[]
Lateral: 42 -> Diogo Camacho Barbosa; 100; 54; 30; 72; 73; 75; 75; (Overall:70,83); Antigos clubes:[]
Lateral: 10 -> Jaime Abreu Fernandes de Oliveira; 75; 18; 41; 60; 80; 54; 57; (Overall:53,70); Antigos clubes:[]
Lateral: 16 -> Catarina Martins Sa Quintas; 84; 52; 66; 50; 69; 86; 62; (Overall:61,38); Antigos clubes:[]
Medio: 4 -> Joao Domingos Pereira Barbosa; 77; 97; 50; 49; 59; 63; 50; (Overall:62,60); Antigos clubes:[]
Medio: 50 -> Arthur Cardoso Franca; 51; 60; 78; 58; 75; 72; 74; (Overall:68,25); Antigos clubes:[]
Medio: 25 -> Leonardo Barbosa Maia; 51; 32; 75; 55; 53; 64; 82; (Overall:62,50); Antigos clubes:[]
Medio: 26 -> Andre Luis Ramos Pereira; 54; 59; 47; 59; 71; 51; 51; (Overall:55,15); Antigos clubes:[]
Medio: 31 -> Tiago Pinheiro da Silva; 63; 75; 78; 82; 54; 93; 64; (Overall:72,85); Antigos clubes:[]

Jogador 2 - Escolha o seu onze inicial(Por numero separado por -enter-)
10
7
5
4
18
42
5
10
17
Numero invalido
14
15
17
11

```

```
Jogador 1 - Escolha 3 jogadores para sair(Por numero separado por -enter-)
[36, 8, 48, 40, 15, 11, 12, 17, 32, 2, 39]
40
15
11
Jogador 1 - Escolha 3 jogadores para entrar(Por numero separado por -enter-)
[20, 21, 22, 23, 25, 27, 28, 29, 30]
22
23
25
```

```
Jogador 2 - Escolha 3 jogadores para sair(Por numero separado por -enter-)
[20, 7, 8, 38, 42, 4, 50, 26, 35, 37, 23]
8
38
42
Jogador 2 - Escolha 3 jogadores para entrar(Por numero separado por -enter-)
[40, 10, 44, 47, 16, 48, 22, 25, 31]
44
47
16
Data: 2021-06-10
Mahler Athletic 1 - 5 Sporting Club Chopin
Equipa Casa: [48, 32, 17, 2, 36, 22, 39, 23, 8, 25, 12]
Equipa Fora: [16, 50, 35, 20, 4, 37, 7, 23, 26, 44, 47]
Substituições casa: {40 -> 22; 11 -> 25; 15 -> 23; }
Substituições Fora: {8 -> 44; 42 -> 16; 38 -> 47; }
```

Simular reencontro

A Liga POO armazena todos os jogos que já ocorreram e esta opção permite ao utilizador repetir um encontro que já tenha acontecido utilizando as mesmas equipas titulares e substituições. Este reencontro é adicionado aos jogos da Liga POO.

```
Opção: 2
0 - Sporting Club Shostakovich 0 - 0 Mendelssohn F. C. -->Data: 2021-03-30
1 - Mozart F. C. 2 - 2 Sporting Club Dvorak -->Data: 2021-04-02
2 - Debussy Athletic 5 - 2 Stravinsky Athletic -->Data: 2021-03-14
3 - Schumann Athletic 2 - 1 Beethoven F. C. -->Data: 2021-02-05
4 - Mendelssohn F. C. 1 - 0 Bartok F. C. -->Data: 2021-02-17
5 - Bach F. C. 1 - 3 Sporting Club Shostakovich -->Data: 2021-02-02
6 - Beethoven F. C. 0 - 2 Handel Athletic -->Data: 2021-01-01
7 - Wagner Athletic 1 - 2 Stravinsky Athletic -->Data: 2021-01-29
8 - Bartok F. C. 0 - 1 Wagner Athletic -->Data: 2021-01-04
9 - Sporting Club Shostakovich 1 - 0 Stravinsky Athletic -->Data: 2021-01-01
10 - Sporting Club Chopin 4 - 0 Sporting Club Schubert -->Data: 2021-03-28
```

```
Numero do jogo para um reencontro:
2
Data: 2021-06-10
Debussy Athletic 3 - 1 Stravinsky Athletic
Equipa Casa: [32, 48, 18, 5, 6, 22, 41, 12, 28, 47, 31]
Equipa Fora: [0, 1, 50, 19, 23, 8, 24, 42, 44, 29, 13]
Substituições casa: {3 -> 47; 21 -> 18; 45 -> 31; }
Substituições Fora: {17 -> 44; 36 -> 23; 22 -> 24; }
```

Consultar Jogador / Equipa / Jogadores sem clube

É permitido o utilizador consultar vários tipos de dados que podem ser úteis para outras funcionalidades.

```
Opção: 1
Nome da Equipa do Jogador:
Mahler Athletic
Numero do Jogador
36
Guarda redes: 36 -> Eduardo Costa de Magalhaes; 58; 45; 94; 36; 6; 59; 47; (Overall:64,64); Antigos clubes:[]
```

```
Opção: 4
Nome da equipa:
Mahler Athletic
Clube: Mahler Athletic (Overall: 59,34)
Avancado: 32 -> Jose Pedro Pinto Teixeira da Cunha; 89; 63; 59; 66; 60; 50; 25; (Overall:61,05); Antigos clubes:[]
Avancado: 2 -> Bruno Filipe Jardim Machado; 90; 73; 56; 60; 41; 72; 53; (Overall:64,90); Antigos clubes:[]
Avancado: 39 -> Guilherme Gil Rocha Goncalves; 68; 31; 13; 65; 30; 61; 71; (Overall:47,45); Antigos clubes:[]
Avancado: 22 -> Pedro Antonio Pires Correia Leite Sequeira; 55; 54; 8; 93; 94; 70; 68; (Overall:66,15); Antigos clubes:[]
Avancado: 28 -> Raul Arieira Parente; 64; 54; 87; 21; 83; 82; 72; (Overall:69,20); Antigos clubes:[]
Defesa: 8 -> Fabio Fernandes Silva; 51; 91; 64; 58; 14; 75; 42; (Overall:54,25); Antigos clubes:[]
Defesa: 48 -> Nelson Ramos Alves; 59; 56; 51; 52; 67; 59; 42; (Overall:53,90); Antigos clubes:[]
Defesa: 27 -> Pedro Miguel Sampaio Faria; 58; 72; 52; 51; 61; 57; 51; (Overall:55,90); Antigos clubes:[]
Guarda redes: 36 -> Eduardo Costa de Magalhaes; 58; 45; 94; 36; 6; 59; 47; (Overall:64,64); Antigos clubes:[]
Lateral: 40 -> Barbara Paulista de Faria; 58; 24; 70; 53; 54; 89; 65; (Overall:46,78); Antigos clubes:[]
Lateral: 15 -> Diogo da Costa e Silva Lima Rebelo; 57; 66; 70; 96; 60; 74; 68; (Overall:57,60); Antigos clubes:[]
Lateral: 20 -> Mariana Dinis Rodrigues; 53; 62; 56; 64; 0; 92; 59; (Overall:50,90); Antigos clubes:[]
Lateral: 25 -> Tomas Cardoso Francisco; 71; 58; 70; 0; 71; 73; 84; (Overall:65,77); Antigos clubes:[]
Lateral: 29 -> Tiago Rafael Fernandes Guimaraes; 61; 58; 58; 61; 37; 95; 55; (Overall:67,25); Antigos clubes:[]
Lateral: 30 -> Pedro Nuno Fernandes Freitas; 53; 71; 63; 45; 5; 63; 66; (Overall:63,70); Antigos clubes:[]
Medio: 11 -> Carlos Daniel da Silva Santos; 56; 75; 84; 63; 65; 56; 69; (Overall:67,00); Antigos clubes:[]
Medio: 12 -> Joao Guilherme Rodrigues Reis; 67; 73; 61; 72; 73; 54; 43; (Overall:51,05); Antigos clubes:[]
Medio: 17 -> Francisco Jose Pereira Teofilo; 68; 66; 96; 74; 69; 57; 76; (Overall:71,25); Antigos clubes:[]
Medio: 21 -> Goncalo Miguel Leao Barros Oliveira Pinto; 59; 3; 1; 88; 52; 54; 72; (Overall:48,70); Antigos clubes:[]
Medio: 23 -> Joao Pedro Martins da Silva Freitas Cardoso; 3; 64; 69; 54; 59; 49; 90; (Overall:59,45); Antigos clubes:[]
```

```
Opção: 5
[Medio: 13 -> Miguel Fernandes; 99; 99; 99; 99; 99; 99; 99; (Overall:99,00); Antigos clubes:[]
]
```

Criar Jogador / Equipa

O utilizador consegue criar equipas e jogadores definidos para estes os parâmetros que quiser.

```
Opção: 6
Nome do Jogador:
Miguel Fernandes
Numero do Jogador:
13
Posicao do Jogador:
Medio
Escolha as habilidades do Jogaodr compreendidas entre 0 e 100
velocidade:
99
resistencia:
99
destreza:
99
impulsao
99
jogo de cabeça:
99
remate:
99
capacidade de passe:
99
Recuperar bolas:
99
Medio: 13 -> Miguel Fernandes; 99; 99; 99; 99; 99; 99; 99; (Overall:99,00); Antigos clubes:[]
```

```
Opção: 7
Nome da equipa:
Grupo 74
Clube: Grupo 74 (Overall: 0,00)
```

Adicionar Jogador a um clube

Dado que existem jogadores ainda sem clube, é permitido ao utilizador pegar nestes e transferi-los para um clube à descrição.

```
Opção: 8
Numero do jogador:
13
Equipa para irá ser transferido:
Grupo 74
Jogador com numero 13 adicionado à equipa Grupo 74 com sucesso
```

Transferir um jogador de um clube para outro

Similar à funcionalidade anterior porém o jogador a ser transferido já está associado a um clube. Esta funcionalidade e a anterior tem o cuidado de verificar se não vão ficar 2 jogadores com o mesmo número, pois o número é a identidade de cada jogador.

```
Opção: 9
Equipa atual:
Mahler Athletic
Numero do Jogador
36
Equipa futura:
Sporting Club Chopin
Transferência do jogador num 36 do Mahler Athletic para o Sporting Club Chopin concluída com sucesso!
```

Conclusão

Em termos de conclusão, achamos que o trabalho se enquadra perfeitamente nos conceitos lecionados e aprendidos nas aulas desta U.C. e permitiu-nos pôr em prática o nosso estudo e desenvolver novas capacidades de programação orientada a objetos.

Embora tal, também temos em conta que o nosso código pode não apresentar a melhor qualidade em alguns pontos(código repetido, inútil ou simplesmente nao aprazível) e que, para além disso, não conseguimos realizar os pontos 7 e 7.1 do enunciado, principalmente por constrição de tempo do que qualquer outro fator.