Instituto Superior de Engenharia de

Coimbra



Departamento de Engenharia Informática e de Sistemas

Engenharia Informática

Programação Orientada a Objetos 2019/2020

Simulador Corridas de Carros

C++

Trabalho Prático - Checkpoint 1

Diogo Marques- 21240125

Miguel Abrantes – 21240373

1. **Quais foram as classes consideradas na primeira versão da aplicação que foi testada?**

A primeira versão tinha as seguintes classes:

* + - Carro
    - Piloto

Estas classes e funcionalidades foram testadas com uma simples função main.

1. **Quais os conceitos/classe que identificou ao ler o enunciado?**

Os conceitos identificados foram:

* Corrida
* Carro
* Piloto (O conceito de polimorfismo devido à existência de vários Pilotos – Não implementado)
* Autódromo
* Pista
* Garagem
* Campeonato //Não sei se corrida e campeonato poderão ser os mesmos xD
* DVG (Classe abstrata que contém todos os tipos de objetos do simulador).

1. **Relativamente a duas das principais classes da aplicação, identifique em que classes ou partes do programa são criados, armazenados e destruídos os seus objetos.**

Autódromo: contém os objetos Pista e Garagem que são criados, assim como destruídos pela mesma.

Equipa: contém os Pilotos e Carros que estejam a agir como uma equipa, ou seja, que o piloto esteja no carro. Ambos os seus objetos são criados e eliminados na classe DVG, independentemente da classe Equipa.

1. **Indique um exemplo de uma responsabilidade atribuída a uma classe que esteja de acordo com a orientação dada acerca de *Encapsulamento*.**

A responsabilidade “criar equipa” está atribuída à classe DVG, porque tem a acesso a todos os pilotos e carros

1. **De entre as classes que fez, escolha duas e justifique por que considera que são classes com objetivo focado, coeso e sem dispersão.**
2. **Relativamente à aplicação entregue, quais as classes que considera com responsabilidades de interface com o utilizador e quais as que representam a lógica?**
3. **Identifique o primeiro objeto para além da camada de iteração com o utilizador que recebe e coordena uma funcionalidade de natureza lógica?**
4. **A classe que representa a envolvente de toda a lógica executa em pormenor muitas funcionalidades, ou delega noutras classes? Indique um exemplo em que esta classe delega uma funcionalidade noutra classe.**
5. **Dê um exemplo de uma funcionalidade que varia conforme o tipo do objeto que a invoca. Indique em que classes e métodos está implementada esta funcionalidade. (*Não é necessário responder a esta pergunta na meta 1*).**