# Resumo:

Neste relatório está explícito todo o desenvolvimento do projecto *“Racing Manager”* desta unidade curricular.

Neste projecto teve-se atenção em reutilizar código, assim ao realizar-se as declarações das classes, organizou-se em hierarquias. Tornando-se mais fácil fazer inserções de novas classes. Por exemplo, as subclasses da classe abstrata Veículo “herdam” todas as variáveis e métodos da classe Veículo. Este procedimento permite reduzir o tamanho do código em cada uma das subclasses e torna possível inserir uma nova subclasse sem qualquer complicação.

Para cada uma das classes, foram definidas as variáveis de instância, os construtores (construtor vazio, construtor por partes e construtor de cópia) e os métodos: *getters, setters*, *equals*, *toString*, *clone*.

Foram feitos outros métodos necessários a cada classe e que permitiram ao utilizador um bom aproveitamento do programa.

Numa última fase foi feito um menu com todas as funcionalidades do programa.

# Introdução

No âmbito da cadeira de Programação Orientada aos Objectos, perante o problema apresentado iremos desenvolver em Java uma aplicação que visa simular acontecimentos desportivos, neste caso mais direccionado para o desporto automóvel.

O objectivo é criar um programa, em que os utilizadores registados no mesmo fazem apostas relativas à classificação de uma prova automobilística que o software vai simular.

Definiu-se as classes **Campeonato, Circuito, Corrida, Aposta, Piloto, Jogador, Veículo**, assim como as suas subclasses. Assim para **Veículo** definiu-se os seguintes tipos: PC1, PC2, GT e SC. Sendo que cada uma destas subclasses “herda” as variáveis de instância e os métodos da classe **Veículo**, poupando-se código e tornando-se mais fácil a reutilização do código.

Criamos também a classe **Manager,** onde disponibilizamos os métodos de interacção com o programa.

Foi também construído um menu para facilitar a utilização dos métodos anteriormente referidos, e tornar o programa mais funcional e interessante.

# 5. Conclusão

Neste projecto um dos objectivos foi a hierarquia das classes para assim haver reutilização do código e facilitar inserções de novas classes e subclasses.

Com o código feito desta maneira evita-se repetição de código, já que o que é comum a alguns tipos encontra-se na classe principal, por exemplo, a informação comum a todos os veículos encontra-se na classe Veiculo e o que é característicos de cada tipo de veículo é especificado em cada subclasse. Também assim se torna mais fácil a leitura.

Para guardar informação utilizou-se HashMaps, HashSets, TreeMaps e o ArrayList. Esta decisão foi bastante importante pra o grupo, pois uma boa escolha permite uma dimensão bastante grande do programa.

Uma etapa realizada com sucesso foi a criação dos métodos que tornaram possível a utilização do programa. Com isso realizado, o programa possuí uma grande variedade de funcionalidades que o utilizador pode tirar proveito.

A interface final, menu, oferece ao utilizador uma utilização fácil e prática fase as funcionalidades que eram esperadas do programa.