

# Semana 1 Revisões, orientação para objectos e módulos

## Exercício - Playlist

Crie e teste uma classe que permita guardar uma lista de músicas e a duração de cada uma em segundos.

## Exercício - Estacionamento

Crie um main() de teste para duas classes que são usadas em conjunto o ParqueDeEstacionamento e o RegistoDeCarro. Pretende-se simular uma parte do software que controla a entrada de carros num ParqueDeEstacionamento pré-pago.

Crie 3 registos de carro com matrículas diferentes e números iniciais de entradas a 0. Crie uma operação para adicionar entradas aos carros (que será chamada quando o dono do carro pagar entradas, por exemplo, num multibanco, ou na banca do vigilante do parque, pode comprar até 10 entradas). No teste simule a compra de uma nova entrada em dois dos carros criados. Insira os carros no registo de acesso ao parque. Simule a tentativa de entrada de todos os carros, bem como de um carro que não consta do registo: apenas os carros que constam do registo e têm um número de entradas superior a zero devem conseguir entrar no parque. As tentativas falhadas devem emitir uma mensagem de erro. Volte a adicionar entradas a todos os carros registados, de maneira a que uma nova tentativa de os fazer entrar no parque tenha sucesso.

Deve ser possível criar um Carro, guardando a matrícula e inicializando o número de entradas no parque de estacionamento a zero. Esse registo deve ser guardado num ParqueDeEstacionamento que guarda todos os carros que têm acesso ao parque (no máximo 100).

Deve ser possível procurar um Carro no ParqueDeEstacionamento pela matrícula.

Deve ser possível também saber se ainda tem entradas pagas, registar a entrada de um carro (subtrair um ao número de entradas pagas) e adicionar um número de entradas pagas dado como argumento (quando o utilizador paga mais entradas).

## Extra: Simulação de Autocarro/Passageiros

Crie um programa que simule uma viagem, num autocarro com lotação de dez pessoas, com origem no Porto e paragens sucessivas em Aveiro, Coimbra e Lisboa. Do Porto arrancam 6 passageiros com destino a Lisboa, e em Aveiro tentam entrar mais 5 passageiros, dois para Coimbra e três para Lisboa. Finalmente, em Coimbra apresentam-se 4 passageiros para Lisboa.

Apesar de serem usadas num caso muito concreto, queremos que as classes desenvolvidas tenham alguma flexibilidade. Pretende-se desenvolver um conjunto de classes que permita simular o fluxo de passageiros durante uma viagem de autocarro. A viagem de autocarro tem uma rota predefinida com paragem em certas cidades. As paragens possíveis serão necessariamente um subconjunto das cidades onde a empresa opera. Neste caso, suponhamos que essas cidades são Lisboa, Coimbra, Aveiro, Porto, e Braga.

Nesta simulação os passageiros são entidades anónimas. Apenas é necessário saber a sua paragem de origem e a sua paragem de destino. Quanto aos autocarros, estes têm um determinado número de lugares e uma rota definida por uma lista ordenada de paragens, sendo a primeira o ponto de partida e a última o destino final. Deverá ser possível saber quantos passageiros estão dentro do autocarro, qual a última paragem efectuada, e quantos passageiros vão sair em determinada paragem.

A simulação deverá poder ser executada com base num conjunto de passageiros (com origem/destino) e um autocarro (com a rota definida). O autocarro segue de paragem em paragem, sendo que em cada paragem saem e entram passageiros de acordo com a sua origem/destino. A simulação deverá emitir mensagens sobre cada paragem, nomeadamente, quantos passageiros saíram, quantos entraram, e quantos não tiveram lugar para viajar.

Tente resolver o problema invocando o mínimo de métodos dos passageiros no teste (os métodos dos passageiros são principalmente invocados dentro de outras classes).