



Nome:

Número:

**DESENVOLVIMENTO DE
SISTEMAS DE SOFTWARE****Exame de recurso**

Lic. Engenharia Informática, Universidade do Minho

17 de janeiro, 2024 – Duração máxima: 2h

Instruções:

Assinale as suas respostas com ■. Não se esqueça de preencher o nome e número. Indique também o número na tabela à direita, assinalando um dígito por coluna.

Leia todo o exame com atenção!

(a cotação mínima das questões é 0 valores)

☐0 ☐0 ☐0 ☐0 ☐0 ☐0☐1 ☐1 ☐1 ☐1 ☐1 ☐1☐2 ☐2 ☐2 ☐2 ☐2 ☐2☐3 ☐3 ☐3 ☐3 ☐3 ☐3☐4 ☐4 ☐4 ☐4 ☐4 ☐4☐5 ☐5 ☐5 ☐5 ☐5 ☐5☐6 ☐6 ☐6 ☐6 ☐6 ☐6☐7 ☐7 ☐7 ☐7 ☐7 ☐7☐8 ☐8 ☐8 ☐8 ☐8 ☐8☐9 ☐9 ☐9 ☐9 ☐9 ☐9

Parte 1 (15 valores)

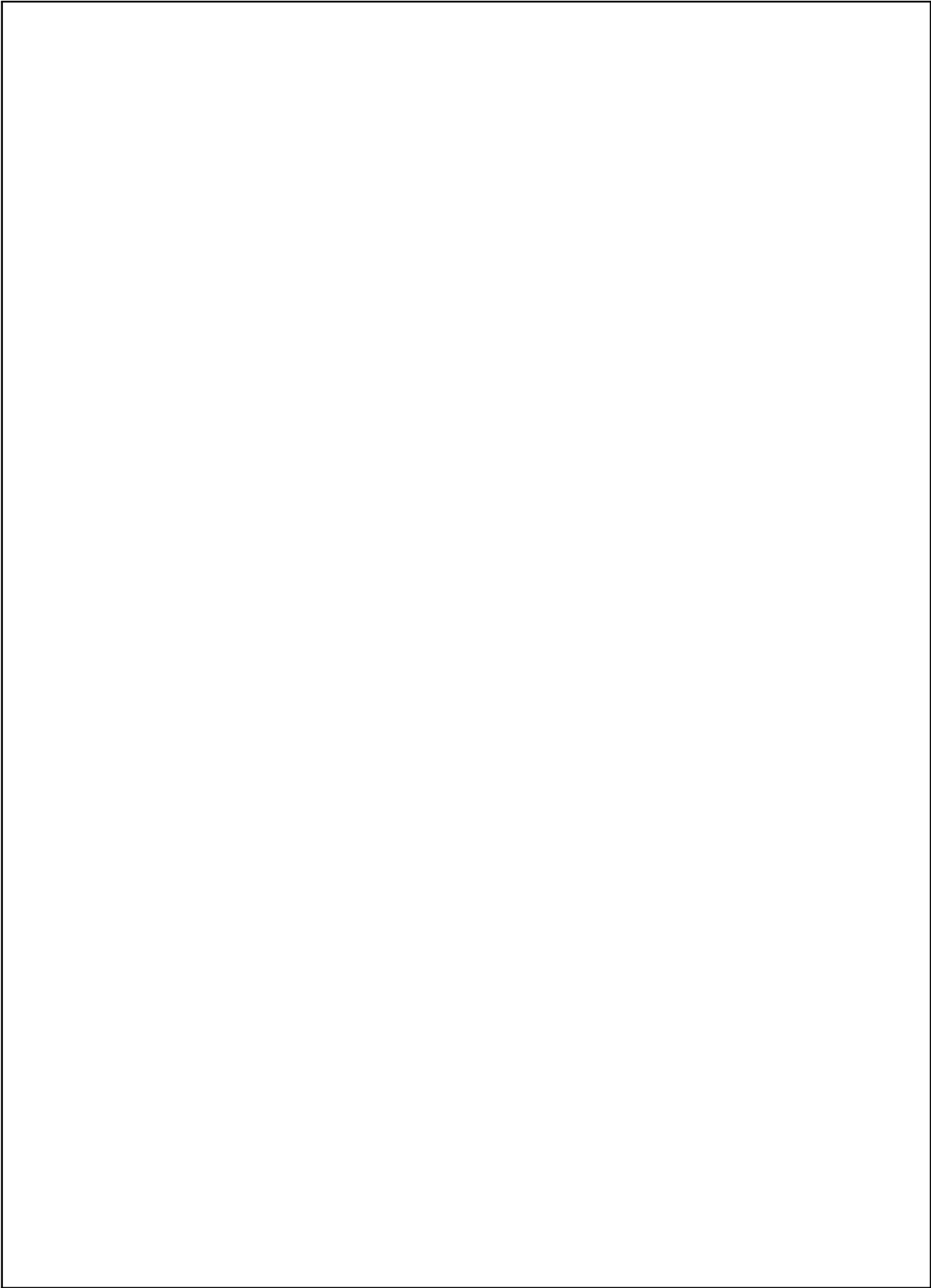
Responda às seguintes questões **em folhas separadas** (note que cada questão pode incluir vários pedidos, responda a todos):

Questão 1 Considere que se pretende modelar um sistema de cobrança automática de lugares de estacionamento. Neste sistema os utilizadores, para que possam estacionar nos diversos parques de estacionamento, devem possuir um dispositivo electrónico que tem um código e que identifica a viatura. Caso um utilizador tenha mais do que uma viatura é necessário que adquira tantos dispositivos electrónicos quantas as viaturas que possui. Sobre cada viatura ficam ainda registados um código identificativo do modelo e a matrícula.

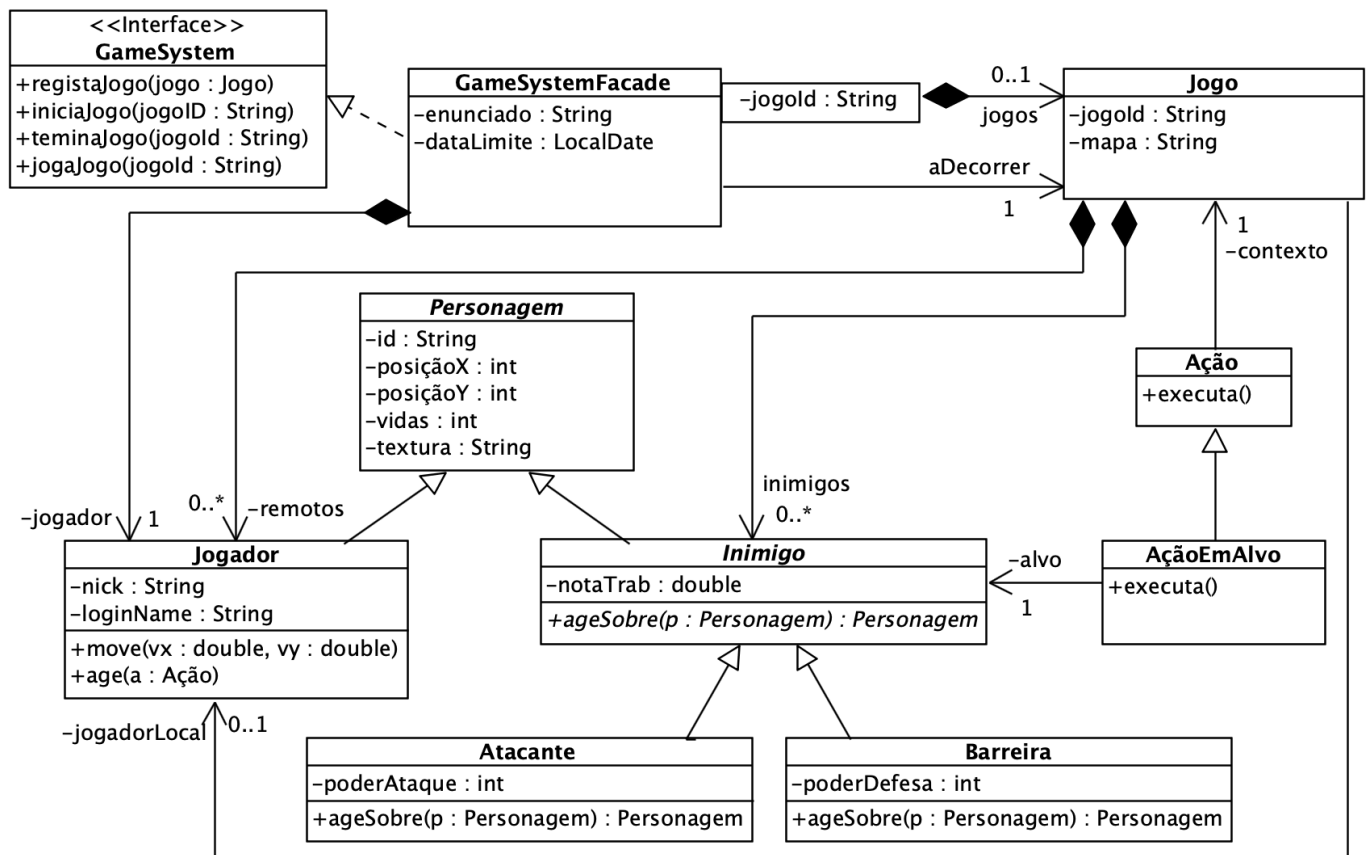
Sempre que um identificador é detectado numa cancela de entrada de um parque de estacionamento, é efectuado um registo de utilização onde ficam registadas a hora e data de entrada, qual o identificador detectado e qual a cancela que originou o registo. Sempre que um carro sai de um parque de estacionamento é detectado pela cancela de saída (consideremos que os parques tem uma cancela de entrada e outra de saída) e o registo associado ao estacionamento é actualizado com a data de saída.

Cada parque de estacionamento tem uma tabela de preços de estacionamento, baseada na classe da viatura. Existem diversas classes de viaturas (moto, carro ligeiro, camião, etc.) e cada parque tem um valor/hora para cada classe de viatura.

Identifique as Entidades e respectivos Relacionamentos contidos na descrição atrás fornecida e desenhe o Modelo de Domínio correspondente. (4 valores)



Questão 2



Considere a arquitetura acima, proposta para a camada de lógica de negócio de um jogo a desenvolver. Cada jogador instala a aplicação no seu computador, registra seus dados (associação jogador de GameSystemFacade para JogadorAtivo) e pode registrar jogos (operação registraJogo (Jogo) e associação jogos). O jogador só pode participar num jogo de cada vez (associação aDecorrer, de GameSystemFacade para Jogo). As operações iniciaJogo e terminaJogo permitem controlar qual o jogo que está a ser jogado.

Cada jogo, além da indicação do jogador que o está a jogar localmente, tem uma lista de jogadores remotos e de inimigos (controlados pelo jogo). Os inimigos podem ser atacantes (que atacam, quer o jogador local ou jogadores remotos, quer outros inimigos), ou barreiras, que impedem o avanço de outros personagens. O efeito da interação entre inimigos e personagens é capturado pela operação ageSobre (p : Personagem).

Os jogadores podem mover-se no mapa do jogo (operação move (vx: **double**, vy: **double**)) e realizar ações (operação age (a: Ação)). As ações são executadas no contexto de um jogo e podem, ou não, ter como alvo um inimigo (ver classes Ação e AçãoEmAlvo). Note que não é mantido um registo das ações executadas. Os objetos que representam as ações existem apenas durante a execução das mesmas.

Considere que lhe foi pedido o desenvolvimento de uma camada de dados para a arquitectura apresentada acima. Sabendo que, a presença de texturas nas personagens recomenda que estas não sejam mantidas em memória mais tempo que o estritamente necessário e que foi decidido mapear toda a hierarquia de Personagem numa única tabela:

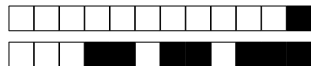
- Indique quais as tabelas que definiria no modelo relacional e redesenhe a arquitectura para incluir DAOs. (4 valores)
- Explique**, de forma breve, quais as razões que levaram a substituir, ou não, cada uma das associações por um DAO. (1 valor)





Questão 3 Considerando novamente a arquitetura que lhe foi proposta na Questão 2, desenhe o diagrama de sequência para o método `iniciaJogo(jogoID: String)` do *facade* que, dado o identificador de um jogo o inicia. Note que, para iniciar um jogo, é necessário verificar que o jogador não seja um dos jogadores remotos do mesmo. Caso seja, o método deve terminar. Caso não seja, o jogador deve ser registado como o jogador local do jogo e a informação sobre o jogo a decorrer atualizada no *facade*. ^(3 valores)

☐ 0 ☐ .1 ☐ .2 ☐ .3 ☐ .4 ☐ .5 ☐ .6 ☐ .7 ☐ .8 ☐ .9 ☐ 1 Reservado aos docentes



Questão 4 Ainda em relação à arquitetura da Questão 2, escreva em OCL o invariante que expressa que nenhum jogo pode ter o jogador local na lista de jogadores remotos. ^(3 valores)
(Considere que associações qualificadas são coleções de valores, tal como feito nas aulas)

☐0 ☐.1 ☐.2 ☐.3 ☐.4 ☐.5 ☐.6 ☐.7 ☐.8 ☐.9 ☐1 *Reservado aos docentes*