

Algoritmos e Estrutura de Dados

Tema 2 – Aluguer de Campos de Ténis (Parte 1)

(23 de novembro de 2018)

Cláudia Mamede up201604832@fe.up.pt
Mariana Neto up201606791@fe.up.pt

Enunciado

Uma empresa municipal possui campos de ténis que disponibiliza aos seus utentes para uso em dois modos: modo de aula (duração de 1h com professor) e modo livre (duração até 2h sem professor: para simplificar, considere períodos múltiplos de 30 minutos). Existe um número máximo de utentes que podem estar em simultâneo nos campos de ténis. As aulas têm prioridade no uso do campo, pelo que o número de utentes admissível em modo livre em cada período é igual a $(\text{NumMaximoUtentes} - \text{NumUtentesAula})$.

O preço de uso do campo de ténis é: fixo, em modo aula; e fixo por período de 30 minutos em modo livre. A empresa possui um cartão *gold* com um custo fixo mensal que permite o acesso a aulas com 15% desconto no valor unitário da aula.

O sistema a implementar deve registar todas as ocorrências em que um utente usa os campos, incluindo a data. No final do mês, o utente deve pagar o uso dos campos nesse mês, sendo-lhe entregue um documento onde são discriminadas todas as datas/horas em que frequentou os campos.

Os utentes que usaram os campos em modo de aula têm ainda acesso a um relatório mensal sobre o seu progresso elaborado pelo professor. As aulas são lecionadas pelos professores existentes, sendo o professor de determinada aula escolhido de modo a equilibrar o número de aulas por professor.

O sistema deve permitir a consulta da ocupação dos campos de ténis, frequência de utentes, contas dos utentes, aulas lecionadas por professores, horário de aulas, horário de professores.

Proposta de resolução

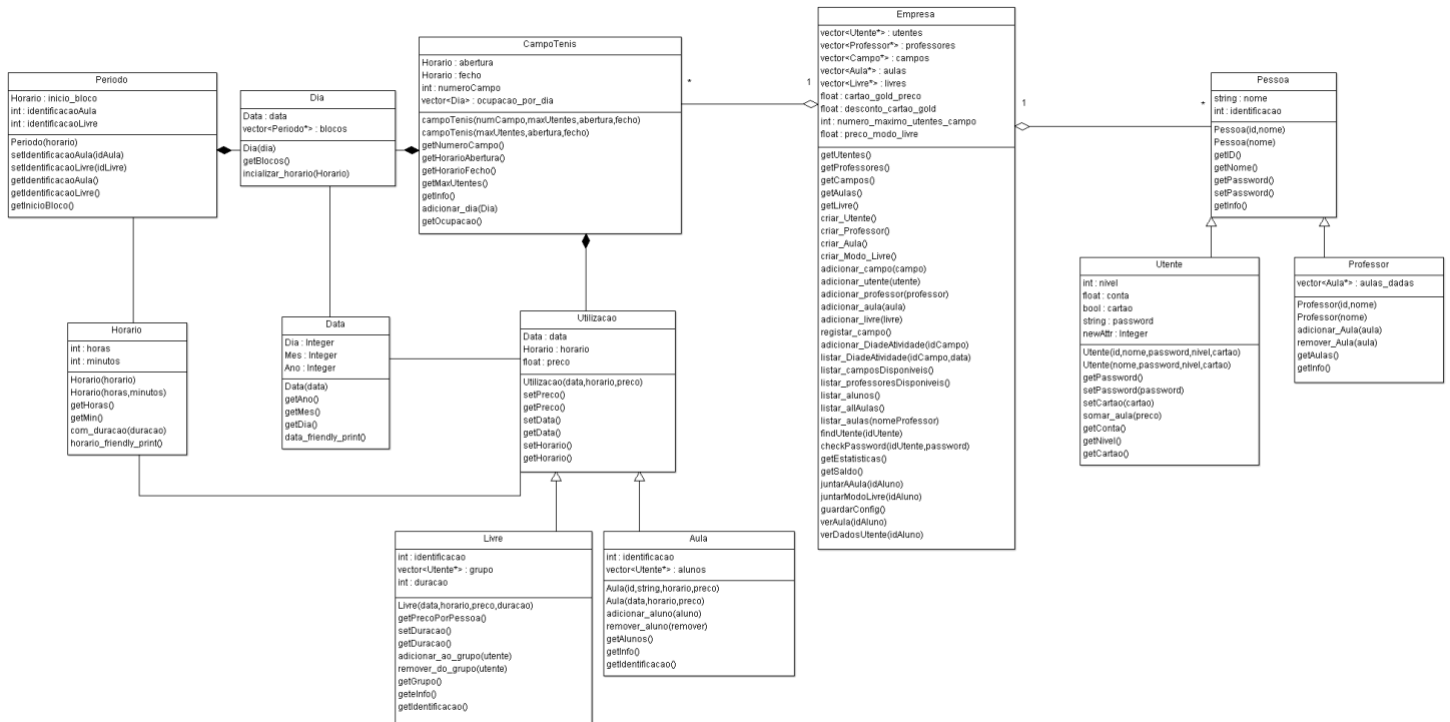
Analisando o enunciado, existem claramente palavras que saltam à vista, nomeadamente “empresa”, “campo”, “aula”, “modo livre”, “utentes” e “professores” pelo que estas foram as primeiras classes a ser implementadas. Aula e Modo livre derivam de Utilização (visto que ambos os modos de utilização possuem uma data, horário e preço)

Para conseguirmos organizar o calendário de funcionamento dos campos, tivemos de criar duas classes extras (para além das Data e Horário): a classe Dia e a classe Período que permitem, de uma maneira eficaz, atribuir aulas e utilizações em modo livre a cada campo. Cada período de 30 min só pode estar ocupado por uma Aula e um Modo Livre (não é possível existirem duas aulas e nenhum modo livre, ou o contrário). Deste modo, sempre que queremos criar uma Aula ou utilização em Modo Livre devemos garantir que primeiro existe um Dia de funcionamento e depois são analisados os Períodos de todos os campos que estão a funcionar nesse Dia de maneira a atribuir uma Utilização para um campo com disponibilidade de Horário.

A atribuição de professores às aulas é feita com recurso a um algoritmo que calcula e analisa o número de aulas de cada professor e retorna aquele que tiver menos número de aulas dadas.

A atribuição de campos às aulas ou utilizações em modo livre é feita com recurso a um algoritmo que encontra o primeiro campo disponível para o dia e períodos em específico (cada aula ocupará 2 períodos e cada utilização em modo livre poderá ocupar entre 1 e $5 \times$ períodos).

Diagrama de classes UML



Nas seguintes classes estão ainda implementados os métodos:

- **Aula:** `bool operator==(const Aula &a2);`
- **Data:** `bool operator<(const Data &outra), bool operator==(const Data &outra);`
- **Pessoa:** `bool operator==(const Pessoa &p1);`
- **Horario:** `bool operator==(const Horario &outroHorario), bool operator>(const Horario &outroHorario);`

As classes

As classes baseiam-se no diagrama UML em cima representado. Por uma questão de simplificação, nas descrições seguintes não são mencionadas as funções do tipo get, set ou operadores uma vez que são todas muito semelhantes e bastante simples.

Empresa

Esta classe pode ser considerada a classe principal uma vez que garante toda a gestão e organização das restantes classes. Tem como atributos privados 5 vetores:

- `vector<Utente*> utentes`: todos os utentes registados no sistema;
- `vector<Professor*> professores`: todos os professores registados no sistema;
- `vector<campoTenis*> campos`: todos os campos registados em sistema;
- `vector<Aula*> aulas`: todas as aulas registadas em sistema;
- `vector<Livre*> modo_livre`: todos as utilizações sem professor (em modo livre) registadas em sistema.

Possui ainda 4 atributos públicos (static) que indicam o valor do cartão gold, o desconto associado ao cartão (15%), o número máximo de utilizadores por campo e o preço da utilização em modo livre. Optou-se por não atribuir um preço fixo às aulas, podendo, deste modo, ser atribuído qualquer valor na altura da criação da aula.

Em termos de funções, a classe Empresa está responsável por registar utentes e professores, criar aulas e utilizações em modo livre, registar campos, atribuir campos às aulas conforme as disponibilidades do espaço e ainda atribuir professores às aulas (esta atribuição visa sempre a atribuição de uma aula ao professor menos ocupado, ou seja, ao professor com menor número de aulas dadas).

Esta classe possui também duas funções de extrema importância nomeadamente:

- `int guardarConfig()`: guarda em ficheiros as alterações feitas no sistema, como por exemplo a adição de utentes ou criação de campos, para que, ao arrancar o sistema, todas as configurações sejam repostas.
- `int inicializar_empresa()`: lê dos ficheiros as informações para criar todos os objetos.

Todas as listagens pedidas no enunciado (e outras) são também realizadas nesta classe.

Utilização

A classe Utilização é a classe base das classes Aula e Livre. Todas as utilizações têm uma data, horário e preço associado.

Aula

A classe Aula é uma classe derivada e tem como atributos a sua identificação (número sequencial que começa em 0 – utilizou-se um atributo estático) e um vetor que mantém o registo dos alunos que realizaram a aula. Nesta classe estão implementados os métodos que auxiliam a adição e remoção de alunos das aulas.

Livre

A classe Livre é uma classe derivada e representa o modo de utilização livre e, à semelhança da classe Aula, possui uma identificação, uma duração (pode variar de 1 a 5, correspondente a blocos de 30min de utilização do campo) e um vetor de Utenes que guarda os utilizadores daquela utilização. Nesta classe também são implementados métodos que auxiliam a adição e remoção de um Utente ao grupo.

Pessoa

Esta classe é a classe base das classes Utente e Professor. Tem como atributos nome e a um número de identificação (semelhante aos outros explicados anteriormente).

Utente

Esta classe é a classe derivada de Pessoa e tem como atributos o nível do utente, a sua conta, uma password e uma variável do tipo *bool* que indica se o Utente em questão tem cartão *gold* ou não.

Professor

Esta classe é a classe derivada de Pessoa e tem como atributos um vetor de Aulas com todas as aulas dadas por esse professor. Estão implementados métodos que permitem adicionar ou remover uma Aula desse vetor.

CampoTenis

A classe campoTenis possui um horário de abertura e fecho, uma identificação (semelhante à utilizada nas classes Aula e Livre) e um vetor de Dias com todos os dias em que o campo estará em funcionamento. A classe tem um método que permite adicionar Dias ao vetor.

Dia e Periodo

Estas duas classes foram a chave para a resolução do problema da gestão de Aulas em diferentes horários e dias.

Cada Dia terá uma Data e um vetor de Periodos (que representam blocos de 30min). A classe Dia possui um método que inicializa o vetor de Periodos, colocando tantos blocos quantos os necessários para preencher um dia de funcionamento de um campo.

Cada Periodo possui um Horario de inicio de bloco e a identificação da aula ou modo de utilização livre que esta a decorrer neste período (0 significa que o campo não está a ser utilizado naquele momento).

Data e Hora

Estas duas classes representam, respetivamente a Data (formato DD-MM-AAAA) e as Horas (formato HHhMM).

Principais dificuldades

As maiores dificuldades surgiram na gestão dos dias e horários de funcionamento dos campos pelo que demoramos algum tempo até conseguir fazer as marcações das aulas e utilizações em modo livre de maneira correta, o que atrasou todo o projeto.

Posto isto, consideramos que o trabalho foi bem conseguido uma vez que conseguimos utilizar a grande maioria dos conteúdos lecionados.