

Programação Modular
Documentação

Trabalho Prático 3

Gerenciador de Campeonatos Esportivos

Alunos

Andre Taiar Marinho Oliveira
Gustavo Henrique Alves Pereira

Professor
Douglas Macharet

Departamento de Ciência da Computação
UFMG - UNIVERSIDADE FEDERAL DE MINAS GERAIS
Belo Horizonte, Minas Gerais

Junho / 2015

Sumário

1	Introdução	1
2	Implementação	1
2.1	Regras	2
2.2	Decisões de Implementação	2
2.3	Execução	4
3	Testes	4
4	Conclusão	6
5	Referências	7

1 Introdução

Esta documentação descreve o terceiro trabalho prático da disciplina de Programação Modular. O tema escolhido para o desenvolvimento do trabalho foi um Gerenciador de Campeonatos Esportivos.

É aqui informalmente definido um **campeonato** como uma sequência de enfrentamentos entre agentes, aqui chamados de **times**, com um resultado único de vencedor (**campeão**) obtido a partir dos resultados dos tais enfrentamentos (**jogos**). Este gerenciador seria utilizado por qualquer pessoa ou organização que desejasse criar e/ou monitorar um torneio esportivo.

O grupo se decidiu por este tema devido à interessante possibilidade de trabalho com hierarquias no caso de um campeonato esportivos. A combinação das entidades envolvidas cria desafios de modelagem bastante construtivos, como a interação entre jogadores e equipes, jogos e campeonatos.

O programa permite a criação e manipulação tanto dos campeonatos quanto das entidades que o compõem.

2 Implementação

A ideia inicial da implementação foi de criar uma interface genérica que permitisse o acompanhamento de campeonatos de diversos tipos. A modelagem contou com a ideia básica de que competições esportivas são similares em um alto nível. Assim, existiriam entidades mais abstratas como Equipes, Jogos e Competições, estendidas por equipes de certos esportes, ou mesmo específica para certas competições. Desta maneira foi imaginado o uso do padrão de projeto Factory, mesmo antes de qualquer código concreto.

Lançadas as bases do trabalho foi realizada uma implementação segundo a metodologia *top-down* de um campeonato. Esta implementação contava com o cadastro simples de algumas equipes num modelo de campeonato baseado no Campeonato Brasileiro de Futebol[1], e conforme as simples regras necessárias para a validação do modelo foram sendo implementadas observou-se que a complexidade aumentava. Dado o tempo disponível e o próprio escopo do trabalho decidiu-se por manter o gerenciador específico para uma competição de futebol. É importante destacar, entretanto, que a modelagem é abrangente o suficiente para permitir a gerência de outros tipos de campeonato e outros tipos de esporte.¹

¹Se tal efeito fosse desejado, uma sugestão seria de que um cliente requisitasse um campeonato específico, que seria implementado pela equipe via hierarquia, recompilando o projeto. Outra ideia se basearia no uso de reflexão em Java, o que implicaria num módulo que adicionaria as classes necessárias.

2.1 Regras

A definição das regras, conforme dito acima, era um ponto a ser incluído no programa para cada caso específico, isto é, cada campeonato diferente tem regras específicas embutidas em sua classe, pelo método *Competicao.processaCompeticao()*. As regras implementadas de fato foram baseadas no Campeonato Brasileiro de Futebol.

Conta-se com um número, não-limitado, de equipes. Cada equipe deverá jogar “em casa” e “fora de casa” uma vez contra cada uma das outras equipes, sendo um desses jogos na primeira etapa e o outro na segunda etapa do campeonato. Modelamos “em casa” e “fora de casa” como “A x B”, sendo A o time da casa e B o time de fora. Na implementação isto serve apenas para diferenciar cada jogo específico, mas a intenção era implementar critérios de desempate e melhor representar a dinâmica real do jogo.

Conforme os jogos são processados a tabela, preenchida, pode ser ordenada. A ordenação possui três critérios de desempate, na simplificação feita: quantidade de pontos, quantidade de vitórias e saldo de gols.

2.2 Decisões de Implementação

Conforme o que foi dito, a intenção de utilizar o padrão de projeto Factory, isto é, delegar para as subclasses a instanciação específica, precede a própria implementação. A ideia de que todos os campeonatos específicos trabalham com um vocabulário em comum era interessante o suficiente para que se fizesse uma modelagem baseada nisso.

Um dos pontos centrais da lógica do programa foi o uso do método *compareTo* das células de tabela de campeonato. As regras de desempate foram inicializadas nela, o que se provou bem simples de fazer e de uma grande praticidade.

Foi criada uma interface gráfica com o uso de Swing, conforme visto em sala. Foram implementados alguns padrões que permitiram uma melhor produção na parte da interface gráfica com o Swing, reutilizando componentes e definindo padrões de layout que conseguimos manter por toda a aplicação de forma consistente. Inicialmente se pensou no uso de uma interface web, mas a abstração funciona independente da visualização de fato utilizada.

O mesmo raciocínio valeu para a camada de persistência de dados. Foi utilizada simples serialização de dados, que perde em sofisticação mas tem amplas vantagens de portabilidade (ser um objeto Java, utilizável por qualquer JVM compatível) e simplicidade.


```

/Library/Java/JavaVirtualMachines/jdk1.7.0_75.jdk/Contents/Home/bin/java ...
1a fase:
Clube de Regatas do Flamengo X Clube América Mineiro
2 0
Clube de Regatas do Flamengo X Clube Atlético Mineiro
0 2
Clube de Regatas do Flamengo X Cruzeiro Esporte Clube
3 0
Clube América Mineiro X Cruzeiro Esporte Clube
2 1
Sport Club Corinthians Paulista X Cruzeiro Esporte Clube
4 0
Clube América Mineiro X Clube Atlético Mineiro
0 2
Sport Club Corinthians Paulista X Clube de Regatas do Flamengo
0 3
Cruzeiro Esporte Clube X Clube Atlético Mineiro
1 3
Sport Club Corinthians Paulista X Clube América Mineiro
2 0
Sport Club Corinthians Paulista X Clube Atlético Mineiro
0 4
2a fase:
Clube Atlético Mineiro X Clube América Mineiro
2 1
Clube Atlético Mineiro X Cruzeiro Esporte Clube
2 2

```

Figura 2: Teste - parte 1

2.3 Execução

A execução implementada, como mostrada na apresentação do trabalho, envolve duas partes. A primeira demonstra a lógica do campeonato. Um pequeno campeonato criado no método `main` do projeto, com dados *hardcoded* demonstra as funcionalidades de jogo e tabela, apresentada ao final. Terminada a lógica, mostra-se a interface gráfica, que demonstra as capacidades de persistência de dados e utilização das abstrações criadas.

3 Testes

2a fase:

Clube Atlético Mineiro X Clube América Mineiro
2 1

Clube Atlético Mineiro X Cruzeiro Esporte Clube
2 2

Clube de Regatas do Flamengo X Sport Club Corinthians Paulista
3 1

Clube América Mineiro X Sport Club Corinthians Paulista
2 1

Cruzeiro Esporte Clube X Sport Club Corinthians Paulista
1 0

Clube América Mineiro X Clube de Regatas do Flamengo
1 0

Cruzeiro Esporte Clube X Clube de Regatas do Flamengo
0 1

Cruzeiro Esporte Clube X Clube América Mineiro
0 3

Clube Atlético Mineiro X Sport Club Corinthians Paulista
4 2

Clube Atlético Mineiro X Clube de Regatas do Flamengo
3 1

Imprimindo tabela ao final do campeonato

Clube Atlético Mineiro
P | V | D | E | GF | GC | SG
22 | 7 | 0 | 1 | 22 | 7 | 15

Clube de Regatas do Flamengo
P | V | D | E | GF | GC | SG
15 | 5 | 3 | 0 | 13 | 7 | 6

Clube América Mineiro
P | V | D | E | GF | GC | SG
12 | 4 | 4 | 0 | 9 | 10 | -1

Sport Club Corinthians Paulista
P | V | D | E | GF | GC | SG
6 | 2 | 6 | 0 | 10 | 17 | -7

Cruzeiro Esporte Clube
P | V | D | E | GF | GC | SG
4 | 1 | 6 | 1 | 5 | 18 | -13

Figura 3: Teste - parte 2

4 Conclusão

O projeto final do curso de Programação Modular teve o objetivo de demonstrar a consolidação dos conhecimentos absorvidos durante a disciplina. Nele foram demonstrados o uso de Java como linguagem orientada a objetos e os benefícios decorrentes do uso do paradigma, especialmente no tocante a modelagem.

Especificamente falando, modelagem foi o principal tema do trabalho. Os detalhes de implementação se mostraram não tão importantes quanto a abstração da ideia, os diagramas utilizados para esquematização e os padrões de projeto escolhidos como idioma comum. A conjunção destes fatores confirmou a importância do conteúdo.

Mesmo os problemas encontrados, especialmente no fator de conjunção entre interface gráfica e lógica do programa, foram compreendidos à luz da modelagem. Muito além de um simples problema de não saber por qual caminho seguir, o grupo compreendeu que problemas de agenda influenciaram mais no desenvolvimento, e há a compreensão que o caminho a seguir para a extensão do programa está bem traçado. Deste modo podemos dizer que concluímos com sucesso parte fundamental do projeto.

5 Referências

- [1] Veja o Regulamento Geral de Competições 2015, *Confederação Brasileira de Futebol* <http://www.cbf.com.br/noticias/campeonato/veja-o-regulamento-geral-de-competicoes-2015#.VX8sWmCTpV4>