

Trabalho Prático 1 – Campeonato PM

Valor: 10 pontos

Data de devolução: 15/09/2014

O objetivo deste trabalho prático é familiarizar o aluno com os principais conceitos de Orientação a Objetos e com a programação utilizando a linguagem Java. Deverá ser implementada uma estrutura de classes que permita armazenar e manipular os dados de alguns de alguns esportes.

Instruções:

Considerando os arquivos de entrada apresentados nas próximas seções, o aluno deve criar os atributos e métodos necessários para acompanhar um campeonato (conjunto de partidas) relativo a 6 esportes: Basquete, Vôlei, Futebol, Vôlei de praia, Futebol de areia e Futebol de salão.

Deve ser implementado o método para calcular o vencedor de uma determinada partida. Entretanto, nesse campeonato em especial, as regras para Vôlei de praia e Futebol de areia são diferentes, onde o vencedor é o time com o menor número de pontos.

Todas as informações deverão ser cadastradas no sistema a partir de um grupo de arquivos de entrada que serão disponibilizados na mesma pasta de execução do programa.

Após a inserção de um conjunto de partidas relativas a cada um dos Esportes, o sistema deverá imprimir as estatísticas de acordo com um arquivo de entrada que será informado. O seu sistema deve tratar possíveis casos de exceção, por exemplo, a requisição de estatísticas de uma seleção não realizou nenhuma partida.

Para guardar as informações relativas às coleções, você pode utilizar qualquer uma das classes que implementa a interface `Collection<E>`. Você pode obter mais informações em: <http://docs.oracle.com/javase/7/docs/api/java/util/Collection.html>.

Funcionalidades:

1. Criar objetos representando cada um dos Esportes mencionados
2. Adicionar um conjunto de Seleções relativo a cada um dos Esportes
3. Adicionar um conjunto de Atletas a cada uma das Seleções
4. Criar Partidas entre as Seleções para cada um dos Esportes
5. Imprimir estatísticas gerais de um determinado Esporte
6. Imprimir estatísticas gerais de uma determinada Seleção

Arquivos de Entrada e Saída:

Exemplo dos arquivos de entrada:

- **Esportes: id;nome**

esportes.txt

```
1;Basquete
2;Volei
3;Futebol
4;Volei de praia
5;Futebol de areia
6;Futebol de salão
```

- **Seleções: id;idEsporte;nomePais**

selecoes.txt

```
1;1;Japão
2;1;USA
3;1;Argentina
4;2;Rússia
5;3;Brasil
```

- **Seleções: id;idSelecao;nome**

atletas.txt

```
1;5;Hulk
2;5;Fred
3;1;Honda
4;1;Yamaha
5;2;Smith
```

- **Partidas: id;idSelecaoA;idSelecaoB;placarSelecaoA;placarSelecaoB**

partidas.txt

```
1;1;2;50;37
2;1;3;19;31
3;2;3;25;25
```

- **Estatísticas: idTipo;(idEsporte ou idSelecao)**

Tipos:

1: Estatísticas gerais de um Esporte

2: Estatísticas gerais de uma Seleção

estatisticas.txt

```
1;1  
2;1
```

Exemplo dos arquivos de saída:

Os arquivos de saída serão relativos apenas às estatísticas. O nome dos arquivos deverá ser composto pelo id do tipo da estatística e o respectivo id do Esporte/Seleção. Considerando o arquivo 'estatisticas.txt' de exemplo acima, os seguintes arquivos devem ser apresentados.

estatistica-1-1.txt

```
Basquete  
Times V      E      D  
Japão 1      0      1  
USA 0        1      1  
Chile 1      1      0
```

estatistica-2-1.txt

```
Japão - Basquete  
Atletas  
Honda  
Yamaha  
Adversários      Placar  
USA               50x37  
Chile             19x31
```

Documentação:

Entre outras coisas, a documentação deve conter:

1. Introdução: descrição do problema a ser resolvido e visão geral sobre o funcionamento do programa.
2. Implementação: descrição sobre a implementação do programa. Devem ser detalhadas as estruturas de dados utilizadas (de preferência com diagramas ilustrativos), o funcionamento das principais funções e procedimentos utilizados, bem como decisões tomadas relativas aos casos e detalhes que porventura estejam omissos no enunciado.
4. Testes: descrição dos testes realizados e listagem da saída (não edite os resultados). Você pode propor outros testes além dos fornecidos com o enunciado.
5. Conclusão: comentários gerais sobre o trabalho e as principais dificuldades encontradas em sua implementação.
6. Bibliografia: bibliografia utilizada para o desenvolvimento do trabalho, incluindo sites da Internet se for o caso.

O que deve ser entregue:

Envie um arquivo ZIP com o nome no formato 'tp1-nome1-nome2.zip', contendo os seguintes arquivos:

- Arquivo README com os nomes completos dos alunos da dupla.
- O código fonte do programa em, Java bem indentado e comentado. Deve ser fornecido junto com o fonte um arquivo Makefile com as opções 'make' e 'make run'.
- A documentação do trabalho bem escrita e detalhada.

Comentários Gerais:

- Comece a fazer este trabalho logo, enquanto o problema está fresco na memória e o prazo para terminá-lo está tão longe quanto jamais poderá estar.
- Clareza, indentação e comentários no programa também serão avaliados.
- O trabalho poderá ser feito em dupla.
- Trabalhos copiados serão penalizados conforme anunciado.
- Penalização por atraso: $(2^d - 1)$ pontos, onde d é o número de dias de atraso.

Critérios de avaliação:

- Funcionamento correto (3 pts).
- Uso correto dos conceitos de OO (5 pts).
- Documentação (2 pts).