**Java Foundations**

**Exercícios - Seção 8: A Liga de Futebol**

Visão Geral

Esse foi um inverno extremamente frio e com muita neve. Nenhum dos nossos amigos queria jogar futebol. Mas agora que a primavera chegou outra temporada da liga pode começar. Seu desafio é criar um programa que modele uma liga de futebol e controle as estatísticas da temporada.

A liga é composta de quatro equipes. As partidas são determinadas aleatoriamente. Toda terça-feira tem dois jogos; assim, todas as equipes jogam semanalmente. Não há um número definido de jogos por temporada. A temporada continua até o inverno chegar.

A liga é muito sensível à temperatura. Os jogadores da defesa ficam muito preguiçosos nos dias quentes. Então, nos dias mais quentes, é possível fazer mais gols durante um jogo. Se a temperatura estiver congelante, nenhum jogo será marcado naquela semana. Se houver três semanas consecutivas de temperaturas muito baixas, isso indica que o inverno chegou, e a temporada está encerrada.

Tarefas

Crie um programa que modele uma liga de futebol e controle as estatísticas da temporada. Considere quais dados devem ser armazenados em uma matriz e quais devem ser armazenados em uma ArrayList. Projete classes com campos e métodos com base na descrição da liga. Você também precisará de uma classe de teste que contenha um método principal. Todos os campos devem ser private. Forneça os métodos getters e setters necessários.

**Equipes**

Cada equipe tem um nome. O programa deve manter o controle do total de vitórias, derrotas, empates,   
gols marcados e gols sofridos de cada equipe. Crie uma matriz que o programador gerenciará.

Imprima as estatísticas de cada equipe quando a temporada terminar.

**Jogos**

Em um jogo, é importante observar o nome e a pontuação de cada equipe, assim como a temperatura do   
dia. Numere cada jogo com um número de identificação inteiro. Esse número aumenta à medida que o jogo é executado. Mantenha controle de cada jogo jogado nessa temporada. Essa classe armazena uma ArrayList de todos os jogos como um campo.

Seu programa deve determinar as pontuações aleatoriamente. O número máximo de gols que qualquer equipe pode armazenar deve aumentar proporcionalmente com a temperatura. Mas certifique-se de que esses números sejam razoáveis.

Quando a temporada terminar, imprima as estatísticas de cada jogo. Imprima a temperatura mais quente e a temperatura média da temporada.

**Programador**

Aceite a entrada do usuário por meio de JOptionPane ou de Scanner. Enquanto o aplicativo está sendo executado, peça que o usuário insira uma temperatura. O programa não deve travar devido à entrada do usuário. Se a temperatura estiver quente o suficiente para haver um jogo, programe dois jogos. Os oponentes são escolhidos aleatoriamente. Certifique-se de que as equipes não sejam programadas para jogar contra si mesmas. Se houver três semanas consecutivas de temperaturas congelantes, a temporada estará encerrada.

**Exemplo de Saída:**

|  |
| --- |
| run:  Too cold to play.  Too cold to play.  Too cold to play.  Season is over  \*\*\*\*\*\*\*\*\*RESULTS\*\*\*\*\*\*\*\*\*  Team 1  Wins: 1, Losses: 1, Ties:0  Points Scored: 9, Points Allowed: 9  Team 2  Wins: 1, Losses: 1, Ties:0  Points Scored: 8, Points Allowed: 8  Team 3  Wins: 0, Losses: 1, Ties:1  Points Scored: 6, Points Allowed: 9  Team 4  Wins: 1, Losses: 0, Ties:1  Points Scored: 8, Points Allowed: 5  Game #1  Temperature: 90  Away Team: Team 2, 4  Home Team: Team 4, 7  Game #2  Temperature: 90  Away Team: Team 1, 8  Home Team: Team 3, 5  Game #3  Temperature: 35  Away Team: Team 1, 1  Home Team: Team 2, 4  Game #4  Temperature: 35  Away Team: Team 3, 1  Home Team: Team 4, 1  Hottest Temp: 90  Average Temp:62.5 |
|  |

**Dica**

Pode ser que seu programa entre em um loop infinito durante este exercício. Essa não é uma situação incomum durante o desenvolvimento de software. É necessário que o programa final tenha uma maneira   
de encerrar a si mesmo. Mas, até você implementar esse recurso, seu IDE fornece um botão **Stop**.

