Mini-Teste 2 de Introdução à Programação (Eng Comp.)
Prof Alexandre Mota – 13/05/2019

Questão. Em vários tipos de campeonatos, usa-se uma tabela de pontuação como forma de criar um ranking entre times/competidores. Esta tabela de pontuação tem várias entradas (um vetor), onde cada entrada possui a seguinte estrutura:

- Nome do time: string, Pontuação: inteiro, Vitórias: inteiro, Empates: inteiro, Derrotas: inteiro, Gols a favor: inteiro, Gols contra: inteiro, Saldo de gols: inteiro

Esta tabela é atualizada conforme partidas são disputadas. A estrutura de uma partida é a seguinte:

- Nome do time 1 (jogando em casa): string, Gols do competidor 1: inteiro, Gols do competidor 2: inteiro, Nome do time 2 (jogando fora de casa): string

A partir de uma partida, a tabela de pontuação é atualizada da seguinte forma:

1. Quem ganha, recebe 3 pontos conquistados. No empate ambos recebem 1 ponto conquistado. Quem perde, não ganha pontos;
2. Se um time ganha, seu número de vitórias aumenta em 1 unidade. Se empata, ambos os times aumentam seus números de empates em 1 unidade. E quem perde, aumenta em 1 unidade seu número de perdas;
3. Para finalizar, os gols são atualizados também. Gols a favor é a quantidade de gols que um time conseguiu fazer. Gols contra é a quantidade de gols que o time sofreu. E o saldo de gols é a subtração entre gols a favor e gols contra;

Por exemplo, considere que a tabela de classificação esteja assim:

1. Sport, 10, 3, 1, 0, 4, 1, 3
2. Náutico, 8, 2, 2, 0, 5, 4, 1
3. Santa Cruz, 7, 2, 1, 0, 4, 4, 0
4. Etc.

E ocorrem as partidas: Santa Cruz 1 X 0 Sport, Náutico 1 X 1 Salgueiro. Então a tabela é atualizada para:

1. Sport, 10, 3, 1, 0, 4, 1, 3
2. Santa Cruz, 10, 3, 1, 0, 5, 4, 1
3. Náutico, 9, 2, 3, 0, 6, 5, 1
4. Etc.

O Santa Cruz ficou em 2º lugar porque seu saldo de gols é menor. Se este saldo fosse igual, aí a ordem seria dada pelo número de vitórias, depois número de derrotas, empates, gols a favor e gols contra, usando a lógica esperada.

Faça um programa em C que controle partidas e a tabela. Use alocação dinâmica. Crie as seguintes funções:

1. Partida *lePartidas(int *qtdPartidas): lê partida do teclado, atualiza o vetor dinâmico e o retorna. Só pára de ler, se o nome do 1º time for . (ponto);
2. Pontuacao *atualizaTabela(Partida *partidas, int qtdPartidas, int qtdPontuacao): usa as partidas para atualizar a tabela, reordenando o vetor de pontuação de acordo com as regras acima;
3. Pontuacao *ordenaTabela(Pontuacao *times, int qtdPontuacao): ordena uma tabela de classificação de acordo com as regras acima;
4. void printTabela(Pontuacao *times, int qtdPontuacao): Apresenta a tabela de classificação na tela;
5. void printPartidas(Partida *partidas, int qtdPartidas): Apresenta as partidas digitadas no teclado;

Boa sorte!!!