

**UFMG - Universidade Federal de Minas Gerais**  
**COLTEC - Colégio Técnico**  
**PROF(A):** Virgínia Fernandes Mota  
**Disciplina:** Algoritmos e Estruturas de Dados **Ano:** 2016

**Prova 2 - Terceiro Trimestre - 10 pontos**

- Prova individual, com consulta (aos arquivos listados abaixo) e em silêncio sepulcral.
  - Bibliotecas criadas durante as aulas.
  - Aulas 15 a 20 com respectivos exercícios.
- Desligue o celular, *chat* ou qualquer outro tipo de mídia social. O não cumprimento deste item acarretará em uma penalidade de 5 pontos.
- É proibida a utilização de material de consulta de terceiros, bem como o empréstimo de material.
- Para que o código fonte seja avaliado, é preciso que esteja "rodando" sem problemas.
- Demonstre seu raciocínio passo a passo e comente seu código.
- Questões com soluções copiadas da Internet ou subsidiadas por terceiros serão anuladas.
- Cada erro de português vale uma dedução de 0,5 ponto.
- As questões devem ser entregues via Dropbox: Crie uma pasta AEDS Prova 2 Terceiro Trimestre. Não será aceito nenhuma questão entregue fora do horário da aula.
- Duração da prova: 100 minutos.





1. (10 pontos) Bem vindo ao Sistema Simplista de Campeonato de League of Legends. O campeonato tem um nome e o ano de sua realização. Em cada campeonato existe um número variável de equipes jogando.

Cada equipe possui um nome, número de vitórias, número de derrotas, número de empates, e é formado por 6 jogadores e 1 líder.

Cada jogador possui nome, salário, tipo do personagem e equipe.

Cada líder possui nome, salário, equipe e senha de acesso ao sistema de jogadores.

Implemente o programa modelado e siga as instruções:

- Fique atento às possíveis heranças e possíveis classes abstratas que possam ser criadas.
- Fique atento aos atributos e métodos de cada classe.
- Crie a classe *TestaCampeonato* com seu método **main** da seguinte forma:
  - Crie o Campeonato Mineiro LoL 2016.
  - Crie duas equipes: NoPainNoGain e ApplePen.
  - Crie 6 jogadores: Jogador1 (topo), Jogador2 (caçador), Jogador3 (atirador), Jogador4 (caçador), Jogador5 (topo), Jogador6 (topo).
  - O líder é responsável por criar sua equipe. Ele deve adicionar os jogadores de sua equipe usando uma senha. Um jogador só pode pertencer a uma equipe e uma equipe não pode ter posições repetidas. Ele deve tentar adicionar os jogadores da seguinte forma:
    - \* Líder do NoPainNoGain: Jogador1, Jogador2, Jogador6.
    - \* Líder do ApplePen: Jogador3, Jogador4, Jogador5, Jogador1.
    - \* Caso não seja possível adicionar o jogador, mostre a mensagem de erro com o motivo.
- Suponha que ApplePen e NoPainNoGain já jogaram algumas partidas.
  - ApplePen: 2 vitórias, 0 derrota, 1 empate.
  - NoPainNoGain: 0 vitória, 2 derrotas, 0 empate.
- **EXTRA:** Crie a classe *RelatorioCampeonato* da seguinte forma:
  - Essa classe é responsável por imprimir o estado atual do campeonato, com seu nome, ano e suas equipes (com jogadores e líder).
  - Essa classe também gera a tabela do campeonato com pontuação em ordem decrescente. A pontuação é dada por: 1 ponto para empate, 3 para vitória, 0 pontos para derrotas.
  - Teste os métodos dessa classe no **main** do seu sistema.