

### Segunda prova de Banco de Dados – 2020

#### Diretrizes:

- A prova pode ser feita individualmente, em dupla ou em trio.
- Você deverá entregar a resposta para essa prova em um arquivo .pdf, no sistema e-Disciplinas **até dia 16/12/2020 às 12h (meio-dia)**. Se você está resolvendo a prova em dupla ou em trio, basta que um dos alunos poste o documento no sistema. **CERTIFIQUE-SE SOBRE A INSERÇÃO DO NOME E NÚMERO USP DOS MEMBROS DO GRUPO NO DOCUMENTO DE REPOSTA.**
- Por favor, utilize o formato desta folha de prova (tipo de fonte (calibri), tamanho de fonte (11), espaçamento entre linhas (simples) e margens justificadas) para criar o documento de resposta.
- Se a resolução do seu exercício exigir a apresentação de um desenho ou de uma fórmula, você pode fazê-los à mão e então colar uma imagem no documento. Mas certifique-se de que você está inserindo uma imagem legível, bem feita e que o documento fique bem organizado.
- Se você fez consultas a materiais bibliográficos ou a mídias disponíveis na internet, cite a fonte da informação na construção da sua resposta. Fazer pesquisas e indicar as fontes é positivo. Fazer pesquisas e não indicar as fontes **NÃO É POSITIVO**. CUIDADO: consulte material para elaborar suas respostas, mas **NÃO COPIE O CONTEÚDO** do material para criar sua resposta.
- **MUITO CUIDADO:** as respostas devem ser pessoais (sua/da dupla/do trio). Havendo identificação de plágio de outros materiais ou de plágio entre as respostas entregues por diferentes alunos, as respostas plagiadas serão desconsideradas de ambas as provas (no caso de plágio entre provas).
- **Dê valor ao instrumento de avaliação.** Você só tem a ganhar com isso. Crie o documento com responsabilidade e dedicação. Tudo o que precisamos nesse período em que estamos vivendo é EDUCAÇÃO, RESPONSABILIDADE e HONESTIDADE.

#### Questão única: Desafio de SQL

Imagine que no próximo ano, o BXCOMP contará com uma edição extra, direcionada aos alunos do quarto semestre do curso de Sistemas de Informação: o I BXCOMP<sup>SQL</sup>. Ou seja, nessa versão do campeonato, os desafios de programação serão criados no contexto de um SGBD Relacional e da linguagem de consulta SQL, ou seja, dado um modelo conceitual Entidade-Relacionamento ou um modelo lógico Relacional e dado um contexto de resolução de problemas que considera um desses modelos, os competidores deverão apresentar uma solução em SQL.

Mas, ninguém no curso de Sistemas de Informação já elaborou desafios de programação nesse contexto e cabe a você propor um desafio para que ele seja incorporado ao conjunto de desafios que serão apresentados aos competidores. Esse é o seu desafio: mostrar o seu conhecimento sobre **a aplicação contextualizada de SQL** por meio da elaboração de um desafio de programação e da apresentação de uma ou mais formas (consultas em SQL) de resolver o desafio.

#### IMPORTANTE:

A resolução dessa prova **PODE** ser totalmente baseada no seu trabalho de banco de dados. O próprio desafio a ser proposto pode ser uma das consultas que você criou no seu trabalho. Ou seja, você **PODE** usar o modelo de dados criado no seu trabalho (total ou parcialmente), o script SQL de criação do banco de dados usado no seu trabalho, o script de população do banco de dados usado no seu trabalho e uma consulta criada no seu trabalho. Mas você também **PODE** usar um contexto diferente do contexto do trabalho, se você assim preferir.

#### MUITO IMPORTANTE:

Caso sua dupla ou trio de resolução dessa prova pertença a grupos de trabalhos diferentes, não há problemas em usar o trabalho de um dos integrantes da dupla ou trio.

#### IMPORTANTÍSSIMO:

Eventualmente dois integrantes de um mesmo grupo de trabalho que estão em duplas ou trios diferentes para resolução da prova podem decidir usar a mesma consulta para criar o desafio. Embora isso não seja um problema, as duas provas deverão ser diferentes o suficiente para não caracterizar um plágio de prova.

Algumas coisas precisam ser consideradas para que o desafio seja elaborado:

- a) O tempo de resolução do desafio durante o campeonato é curto. Então, você não deve esperar que os competidores criem os modelos de dados. Você deve fornecer o(s) modelo(s) de dados para eles.
- b) Além disso, os desafios de programação possuem casos de testes pré-estabelecidos, então, o esquema do banco de dados precisa ser estabelecido de antemão, ou não será simples, e talvez possível, criar os casos de testes.
- c) Os modelos dados que representam o esquema do banco de dados podem ser maiores do que o necessário para resolver o problema enunciado. Ou seja, você pode, a seu critério, apresentar um modelo que abrange mais relações do que aquelas que serão usadas pelo competidor na consulta SQL que ele desenvolverá para resolver o problema.
- d) Você não precisa fornecer o esquema do banco de dados considerando o modelo lógico. Você pode oferecer o modelo conceitual e deixar que o competidor mostre o seu conhecimento sobre mapeamento. Porém, se essa for sua opção, informe um padrão de projeto de mapeamento que deve ser seguido e evite envolver casos de especialização-generalização, pois o mapeamento desses casos pode assumir diferentes estratégias e pode confundir o competidor.
- e) Sobre a questão da especialização-generalização, exceções podem ser consideradas se o problema a ser resolvido envolver justamente o conhecimento sobre as diferentes possibilidades de mapeamento. Tudo vai depender da sua criatividade.

Para construir e apresentar o seu desafio de programação você deverá elaborar:

- a) O **enunciado** (um arquivo PDF), no qual você explica qual é o problema a ser resolvido. Nesse enunciado você deve:
  - I. Contextualizar o seu problema, ou seja, contar a história que motiva a construção de uma consulta SQL.
  - II. Definir o problema claramente, apresentando o problema a ser resolvido (o enunciado para a consulta SQL que o competidor deverá criar) e apresentando o modelo de dados sobre o qual a consulta SQL que resolve o problema será executada.
  - III. Apresentar pelo menos um caso de teste (não todos) com a respectiva resposta esperada, para que o competidor possa testar a consulta SQL que ele está fazendo.
- b) O arquivo **teste.in** no qual você fornece entradas para teste do programa. Entradas para o programa são instâncias do banco de dados que dizem respeito ao contexto de resolução do problema. Nesse arquivo deve constar tanto o caso de teste, ou os casos de testes, oferecido para o competidor, quanto pelo menos mais um caso de teste que o competidor não tem acesso.
- c) O arquivo **teste.out** no qual você fornece saídas para os casos de testes do programa. Saídas para o programa são instâncias da tabela temporária criada pela execução da consulta SQL que resolve o problema do seu desafio.

Note que fazer casos de testes diferentes significa testar aspectos diferentes de uma solução. Lembrando dos casos de prova da matemática: provamos para o caso  $<0$ , para o caso  $=0$  e para o caso  $>0$ . Casos de teste para um programa devem seguir o mesmo princípio. Não faz sentido fazer dois casos de teste diferentes para testar exatamente o mesmo aspecto da solução.

Para cumprir com o requisito desta prova, você deverá entregar no sistema e-disciplinas um arquivo compactado contendo:

- a) O arquivo PDF do enunciado do desafio (um arquivo pdf).
- b) O arquivo teste.in (um arquivo txt ou um arquivo pdf).
- c) O arquivo teste.out (um arquivo txt ou um arquivo pdf).
- d) O script SQL de criação do modelo de dados usado no desafio (um arquivo txt).
- e) O script SQL de população desse modelo de dados de acordo com o arquivo teste.in (um arquivo txt).
- f) A consulta SQL que resolve o problema apresentado no desafio (um arquivo txt).

**Boa prova! Sejam dedicados! Sejam criativos! Mostrem o seu conhecimento!**