

# Design e Desenvolvimento de Bancos de Dados I — *Syllabus* —

Prof. Abrantes Araújo Silva Filho

2023/1

## LEIA COM ATENÇÃO!

Além do material oficial que está disponível no Portal do Aluno da UVV (Universidade de Vila Velha), este documento contém mais detalhes sobre a dinâmica da disciplina **Design e Desenvolvimento de Bancos de Dados I** para todas as turmas deste semestre (2023/1). Leia-o com atenção e procure o professor para esclarecer qualquer dúvida. Note que este *syllabus*<sup>1</sup> não é um documento “oficial”: é um material extra, preparado pelo professor da disciplina, para proporcionar informações adicionais importantes para os alunos. A ementa oficial da disciplina está [disponível para download](#).

## Sumário

<b>1</b>	<b>Visão geral da disciplina</b>	<b>3</b>
<b>2</b>	<b>Pré-requisitos</b>	<b>4</b>
<b>3</b>	<b>Objetivos do curso e resultados esperados</b>	<b>4</b>
3.1	Objetivos de aprendizagem . . . . .	4
3.2	Habilidades a serem adquiridas . . . . .	5
<b>4</b>	<b>Estrutura e informações sobre a disciplina</b>	<b>6</b>
4.1	Professor . . . . .	6
4.2	Monitores, assistentes e tutores . . . . .	6
4.3	Datas e horários das aulas . . . . .	6
4.4	Laboratório de computação e computador pessoal . . . . .	7

---

<sup>1</sup>Ementa, plano de ensino, programa de estudos.

4.5	Grau de dificuldade e esforço estimado . . . . .	8
4.6	Ambientes virtuais de aprendizagem . . . . .	9
4.7	Materiais da disciplina . . . . .	10
<b>5</b>	<b>Dinâmica da disciplina</b>	<b>12</b>
5.1	Leituras, trabalhos e atividades . . . . .	13
5.2	Aulas . . . . .	14
5.3	<i>Problem Sets</i> (PSETs) . . . . .	14
5.4	Monitoria e tutoria . . . . .	15
<b>6</b>	<b>Avaliação da aprendizagem</b>	<b>15</b>
6.1	Provas . . . . .	15
6.2	PSETs . . . . .	16
6.3	Trabalhos e atividades . . . . .	16
6.4	Testes rápidos, vídeos etc. . . . .	16
6.5	Composição da nota final . . . . .	16
<b>7</b>	<b>Integridade acadêmica</b>	<b>17</b>
7.1	Política sobre trabalho colaborativo . . . . .	19
7.2	Cláusula de arrependimento . . . . .	20
<b>8</b>	<b>Perguntas freqüentes</b>	<b>21</b>

# 1 Visão geral da disciplina

Olá, seja muito bem-vindo!

Esta é a disciplina **Design e Desenvolvimento de Bancos de Dados I**, para os cursos de **Ciência da Computação** e **Sistemas de Informação** da **Universidade de Vila Velha (UVV)**.

Esta disciplina é uma introdução aos conceitos fundamentais necessários para projetar, usar e implementar sistemas de bancos de dados e aplicações de bancos de dados, através de uma abordagem profunda e atualizada dos aspectos mais importantes dos sistemas e aplicações de bancos de dados, bem como das tecnologias relacionadas.

Nesta disciplina você terá o primeiro contato com os **bancos de dados** e também com os **sistemas de gerenciamento de bancos de dados** *open-source* (tais como o **MySQL**, **MariaDB** e o **PostgreSQL**) e comerciais<sup>2</sup> (tais como o **Oracle Database** e o **Microsoft SQL Server**).

Você aprenderá a projetar e implementar um banco de dados através de técnicas de modelagem conceitual, lógica e física, de tal forma que o banco de dados seja adequado para a tomada de decisões e com boa performance para uso no mundo real. Também estudará o modelo relacional de dados, a linguagem SQL (*Structured Query Language*), álgebra relacional, técnicas de normalização, bancos de dados de objeto, XML, modelo de dados entidade-relacionamento e muito mais!

Além disso esta disciplina também contribuirá para que você desenvolva suas habilidades de **pensamento computacional**<sup>3</sup> quando você estiver aprendendo a modelar bancos de dados: 1) você precisará **decompor** um problema complexo em problemas menores, mais gerenciáveis; 2) você precisará usar técnicas de **abstração** para se focar nas partes essenciais dos problemas, ignorando detalhes irrelevantes; 3) você começará a usar técnicas de **reconhecimento de padrões** para pensar em como resolver problemas; e 4) você começará a entender sobre **algoritmos** e linguagens.

Em resumo, esta disciplina é sua porta de entrada para o fantástico mundo dos bancos de dados.

Prepare-se, pois esta disciplina é **muito extensa**, **ádua** e **difícil**, mas o resultado, se você se dedicar conforme descrito neste plano de ensino, será recompensador!

---

<sup>2</sup>No caso dos sistemas comerciais, utilizaremos a versão gratuita desses bancos de dados, o Oracle Database Express Edition e o Microsoft SQL Server Express.

<sup>3</sup>Para uma discussão inicial sobre pensamento computacional, leia a série de artigos disponíveis no site da *British Broadcasting Corporation* (BBC) em <https://www.bbc.co.uk/bitesize/topics/z7tp34j> (os artigos estão em inglês).

## 2 Pré-requisitos

### Obrigatórios<sup>4</sup>:

- **Matemática:** familiaridade e conforto com, pelo menos, a matemática do ensino médio (aritmética, álgebra, funções e conjuntos). Caso sua base matemática seja fraca, veja o texto “[Preparação para graduação em áreas exatas: matemática além do básico](#)”, com dicas para estudar novamente a matemática, do jeito certo;
- **Computador:** você deve ter um computador com acesso a internet (câmera, microfone e caixas de som são recomendados) e com capacidade suficiente para instalar e executar os softwares que serão utilizados (um computador recente com Linux, Windows ou Mac, com 8 GiB de memória RAM e bastante espaço em disco). Verifique a Seção 4.4 para maiores detalhes; e
- **Inglês:** livros e leituras obrigatórias da disciplina estarão em português mas haverá também bastante conteúdo em inglês (principalmente leitura). Na área da computação é absolutamente fundamental o domínio de inglês (se você não sabe inglês COMECE JÁ a estudar).

### Desejáveis<sup>5</sup>:

- **SQL:** algum curso introdutório em SQL;
- **Programação:** algum curso introdutório de programação, em qualquer linguagem.

## 3 Objetivos do curso e resultados esperados

### 3.1 Objetivos de aprendizagem

- Aprender noções fundamentais sobre bancos de dados (banco de dados, sistemas de gerenciamento de bancos de dados, características de um sistema de bancos de dados, vantagens e desvantagens);
- Aprender o que é modelo de dados, a arquitetura dos bancos de dados, quais as linguagens e interfaces de acesso aos dados, e arquitetura cliente/servidor;

---

<sup>4</sup>São conhecimentos prévios e/ou recursos que você, obrigatoriamente, deve ter para que possa acompanhar adequadamente a disciplina e aprender o conteúdo. Caso você não possua algum dos pré-requisitos obrigatórios poderá ter dificuldade para acompanhar as aulas ou realizar todas as atividades propostas pelo professor (nesse caso converse **imediatamente** com o professor para explicar a situação e tentar encontrar uma solução).

<sup>5</sup>São conhecimentos prévios e/ou recursos não obrigatórias mas que, se você tiver, podem facilitar seu estudo (você já terá uma base a partir da qual irá desenvolver novas habilidades). Se você não tiver nenhum dos requisitos desejáveis, não se preocupe: você conseguirá acompanhar a disciplina.

- Entender o modelo relacional de dados e suas restrições, bem como as operações de atualização, transações e tratamento de violações de restrições;
- Aprender a usar comandos SQL básicos (select, insert, update, delete);
- Aprender a usar *queries* SQL mais avançadas (joins, triggers, views) e alterações no esquema de um banco de dados;
- Entender a relação entre a álgebra e o cálculo relacional com a SQL;
- Aprender a modelar um banco de dados utilizando o modelo entidade-relacionamento (ER);
- Aprender o modelo entidade-relacionamento estendido (EER) para trabalhar com subclasses, herança, especialização, generalização e conceitos relacionados;
- Aprender a mapear um modelo conceitual (ER ou EER) para o modelo relacional;
- Entender dependências funcionais e como normalizar um modelo relacional, sabendo usar diversas formas normais;
- Aprender sobre regras de inferência, equivalência, cobertura, algoritmos e decomposições relacionais;
- Aprender sobre sistemas de gerenciamento de bancos de dados de objeto e objeto-relacional;
- Diferenciar dados estruturados de não estruturados, sabendo utilizar modelos de dados hierárquicos da XML;
- Aprender as técnicas básicas de programação SQL (SQL embutida, SQL dinâmica e SQLJ); e
- Utilizar bancos de dados na web através de um exemplo com PHP.

### 3.2 Habilidades a serem adquiridas

- Projetar e modelar bancos de dados (modelos conceitual, lógico e físico);
- Implementar bancos de dados em diversos sistemas de gerenciamento de bancos de dados (MySQL, MariaDB, PostgreSQL, Oracle e SQL Server);
- Utilizar um sistema Linux para acesso aos bancos de dados, principalmente o uso da linha de comando;
- Aprender as interfaces de linha de comando para acesso aos bancos de dados;
- Aprender uma ferramenta gráfica para acesso a diversos bancos de dados (utilizaremos a [DBeaver](#));
- Aprender uma ferramenta web para acesso a um banco de dados (utilizaremos o [phpMyAdmin](#)); e
- Aprender uma ferramenta gráfica dedicada para acesso a um banco de dados (utilizaremos o [pgAdmin](#)).

## 4 Estrutura e informações sobre a disciplina

### 4.1 Professor

Meu nome é **Abrantes Araújo Silva Filho** e serei o professor responsável pela sua turma. Algumas informações importantes:

 Sou graduado em **medicina** pela **UFES** (1999), mestre em **epidemiologia e métodos quantitativos em saúde** pela **FIOCRUZ** (2002) e graduado em **ciência da computação** pela **FAESA** (2021). Atualmente sou estudante do bacharelado em **matemática** da **Uninter**, na modalidade de ensino a distância (EAD).

 [www.linkedin.com/in/abrantess-filho](http://www.linkedin.com/in/abrantess-filho)

 [github.com/abrantessaf](https://github.com/abrantessaf) e também [github.com/computacaoraiz](https://github.com/computacaoraiz)

 [www.abrantess.pro.br](http://www.abrantess.pro.br) e também [www.computacaoraiz.com.br](http://www.computacaoraiz.com.br)

 [abrantessaf@uvv.br](mailto:abrantessaf@uvv.br)

 Canal **Computação Raiz** em <https://www.youtube.com/computacaoraiz>

 **Cursos do Computação Raiz** em <https://cursos.computacaoraiz.com.br>

 (27) 9-9991-4393, para entrar em contato via **Signal** ou **Telegram**<sup>6</sup>

### 4.2 Monitores, assistentes e tutores

Oficialmente ainda não está confirmada a participação de nenhum monitor, assistente ou tutor. Todas as atividades serão conduzidas diretamente pelo professor, de modo presencial.

Extra-oficialmente, alunos mais avançados ou que já tenham experiência prévia podem ser voluntários para atuar como monitores informais da turma, auxiliando os alunos com menor experiência e/ou dificuldade.

Para mais informações, consulte a Seção **5.4**.

### 4.3 Datas e horários das aulas

O período 2023/1 se inicia no dia 01/fev/2023 e termina no dia 04/jul/2023.

Oficialmente cada turma tem 2 aulas obrigatórias por semana. Além disso, caso um monitor seja oficialmente selecionado pela UVV, as turmas terão à disposição um monitor durante todas as tardes da semana para discussões, esclarecimento de dúvidas, auxílio em exercícios e tarefas e muito mais.

Cada turma também utilizará a plataforma **Cursos do Computação Raiz**, no endereço <https://cursos.computacaoraiz.com.br>. Os alunos serão matriculados no

---

<sup>6</sup>Eu não uso o WhatsApp por motivos de privacidade. Se quiser me mandar uma mensagem, você terá que usar o Signal ou o Telegram.

curso **CR6.180A: Introdução ao Projeto e Sistemas de Bancos de Dados** e deverão assistir aos vídeos, realizar as provas, as atividades, os PSETs (mais sobre isso em breve) e tudo o mais que estiver agendado. **ATENÇÃO:** o curso CR6.180A faz parte da nota para sua aprovação nesta disciplina!

Cada turma deve consultar o site [Disciplinas UVV](#) e verificar seu horário semanal e o cronograma previsto de aulas e conteúdo para a disciplina.

Aulas extras podem ser necessárias para a reposição de aulas canceladas por motivo de força maior. As aulas de reposição são oficiais e devem ser aprovadas por, pelo menos, 75% dos alunos.

#### 4.4 Laboratório de computação e computador pessoal

A estrutura de laboratórios de computação da UVV é excelente, possuindo máquinas com hardware e software padronizados que facilitam a realização das atividades. Além disso o benefício mais importante de trabalhar no laboratório é obter ajuda direta do professor, dos assistentes, e de seus próprios colegas de turma.

Entretanto, por limitações de tempo, você precisará realizar tarefas em seu próprio computador pessoal. Veja aqui as configurações mínimas necessárias para que você consiga realizar todas as tarefas em seu próprio computador:

-  **Processador:** deve ter arquitetura INTEL x86, de 64 bits. Outras arquiteturas como a ARM (MacBook Air M1, por exemplo) não são suportadas. Qualquer computador relativamente recente com processador Intel ou AMD já está adequado;
-  **Memória:** no mínimo 8 GiB de memória RAM (16 GiB é recomendável);
-  **Disco rígido:** pelo menos 50 GiB de espaço livre em disco (70 GiB durante a instalação da máquina virtual do curso);
-  **Sistema operacional:** Linux, Windows ou Mac;
-  **Monitor:** no mínimo um monitor HD (Full HD é recomendável). Seu monitor deve ser grande o suficiente para que você possa trabalhar com conforto. Se você tiver dois monitores poderá realizar as tarefas em um monitor e usar o outro para ler o conteúdo ou consultar a documentação sobre bancos de dados disponível na Internet;
-  **Internet:** você deve ter uma conexão razoavelmente rápida para fazer o download da máquina virtual. Além disso você precisa assistir todas as aulas e realizar as tarefas do curso CR6.180A;
-  **Som:** o uso de caixas de som ou *headphones* é necessário para assistir às aulas do curso CR6.180A;
-  **Microfone:** não é obrigatório, mas pode ser utilizado caso ocorram aulas online síncronas; e

- **Câmera:** não é obrigatória mas uma *webcam* pode ser utilizada caso alguma aula extra seja feita de modo online.

Se você tiver um *notebook* pode utilizá-lo durante as aulas de laboratório, ao invés dos computadores da UVV. A vantagem de usar seu próprio *notebook* é que você terá o mesmo ambiente, tanto na UVV quanto em casa, e poderá seguir os exercícios e tarefas de modo contínuo.

## 4.5 Grau de dificuldade e esforço estimado

Tenha sempre em mente que esta disciplina é extensa e precisa de **dedicação, estudo e leitura diária** para que você consiga aprender tudo o que é necessário. Em especial atente-se para o seguinte:

- **Nível:** esta disciplina é de nível introdutório a intermediário;
- **Língua:** as aulas, livros e materiais da disciplina são em **português**, mas alguns materiais (leituras e vídeos) estão em **inglês**. Os softwares utilizados estão, em sua maioria, em inglês (alguns oferecem tradução para português);
- **Esforço total:** 10–15 horas de estudo por semana, **além da carga horária das aulas presenciais**, dependendo de seu conhecimento e habilidades prévias. Prepare-se para dedicar, em média, 1–2 horas diárias para leitura e realização das atividades programadas;
- **Leitura:** parte do esforço nessa disciplina é voltado para atividades de leitura. Prepare-se para ler, em média, **35 páginas por semana** (variando de 18 a 52 páginas por semana). Não se assuste com a quantidade pois há muitas ilustrações, exemplos e códigos de programação nos livros utilizados, fazendo com que a leitura não seja tão hercúlea assim. Mantenha as leituras em dia, conforme o cronograma. Acumular leituras é “fatal” para o aprendizado; e
- **Trabalhos e atividades:** a maior parte do esforço nessa matéria é representada pelos trabalhos, *PSETs* e atividades práticas que você realizará semanalmente.

Você já percebeu que esta disciplina precisa de um grau de **dedicação de tempo e esforço** consideráveis. Mesmo assim, se você estiver gastando mais tempo do que esforço total indicado acima, pode ser um sinal de que você precisa de ajuda para entender melhor o material. Entre imediatamente em contato com o professor, não deixe acumular dúvidas.

Eu espero que você, como estudante universitário<sup>7</sup>, esteja buscando aprender o máximo possível e fazendo todo esforço para se dedicar nas atividades propostas e aprender por conta própria de acordo com as orientações do professor. Lembre-se:

---

<sup>7</sup>O estudo em nível universitário é **muito diferente** do estudo em nível de ensino médio. Na universidade você terá que ler muito e aprender muita coisa por conta própria. Não espere que o professor vá passar todo o conteúdo mastigado e resumido para você decorar para a prova, essa vida acabou!

a única maneira de aprender é estudar! Leia os textos indicados abaixo para saber o que faz a diferença no momento de estudar:

- [Como estudar? Dicas ao estudante iniciante](#)
- [O problema é o tamanho do lápis](#)
- [Você gasta sua borracha?](#)

Eu também espero que você seja um **estudante** e não um **aluno**. E que você tenha **brio**, seja **bravo**, **persistente** e **resiliente** para encarar o desafio! Assista os pequenos vídeos indicados a seguir para saber a diferença entre aluno e estudante, e para saber a necessidade de brio para estudar:

- ▶ [Aluno ou estudante?](#) Do professor Pierluigi Piazzzi; e
- ▶ [Você tem brio?](#) Do professor Clovis de Barros Filho (desconsidere os palavrões, assista até os final!)

Para saber **como vencer a procrastinação e manter o estudo em dia**, assista ao seguinte vídeo (ative as legendas em português se necessário):

- ▶ [Why we procrastinate?](#) de Vik Nithy.

## 4.6 Ambientes virtuais de aprendizagem

- **CR6.180A: Introdução ao Projeto e Sistemas de Bancos de Dados:** este curso do Computação Raiz é parte integral desta disciplina, de forma OBRIGATORIA (provas e atividades para nota e aprovação serão realizadas no CR6.180A). Este curso contém vídeos, exercícios, fóruns de discussão e outros recursos para o aprendizado. Os alunos serão matriculadas pelo professor neste curso.
- **Oracle Academy:** a UVV fez uma parceria com a Oracle para disponibilizar aos alunos a plataforma Oracle Academy, com diversos cursos da própria Oracle referentes ao conteúdo da disciplina. Você receberá um usuário/senha para entrar na Oracle Academy e estudar os conteúdos disponibilizados pelo professor. Este ambiente é OPCIONAL mas altamente recomendado.
- **Portal do Aluno da UVV:** será utilizado para distribuir a documentação oficial da disciplina, postar mensagens aos alunos, indicar leituras, distribuir materiais, distribuir tarefas e atividades e, de modo geral, para fazer toda a comunicação entre o professor e a turma.

Atenção: se você estiver sem acesso ao Portal do Aluno da UVV, entre em contato com a coordenação do curso.

## 4.7 Materiais da disciplina

📖 **Livro texto de referência:** utilizaremos como livro texto de referência da disciplina (e de onde serão feitas as leituras obrigatórias) o:

- ELMASRI, R. & NAVATHE, S. B. **Sistemas de Banco de Dados**. 7ª edição. São Paulo: Pearson, 2018.

Observação: se você não conseguir encontrar a 7ª edição do ELMASRI, pode substituir por edições anteriores sem problemas (a biblioteca da UVV conta com exemplares da 4ª e 6ª edições).

Atenção: você **não precisa comprar**<sup>8</sup> o livro, utilize os exemplares disponíveis na biblioteca.

📖 **Outros livros recomendados:** para os alunos que quiserem se aprofundar e complementar a leitura do livro texto de referência, os seguintes livros podem ser consultados:

- SILBERSCHATZ, A.; KORTH, H. F.; SUDARSHAN, S. **Sistema de Banco de Dados**. 7ª edição. Rio de Janeiro, 2020.
- CORONEL, C; MORRIS, S. **Database Systems: design, implementation, & management**. 13th edition. Boston: Cengage, 2019.
- RAMAKRISHNAN, R; GEHRKE, J. **Sistemas de Gerenciamento de Banco de Dados**. 3ª edição. São Paulo: McGraw-Hill, 2008.
- DATE, C. J. **An Introduction to Database Systems**. 8th ed. Boston: Pearson, 2004.
- HERNANDEZ, M. J. **Database Design for Mere Mortals: a hands-on guide to relational database design**. 25th edition. Boston: Pearson, 2021.
- KYTE, T. **Effective Oracle by Design**. 1st edition. New York: McGraw-Hill/Osborne, 2003.
- VIESCAS, J. L. **SQL Queries for Mere Mortals: a hands-on guide to data manipulation in SQL**. 4th edition. Boston: Pearson, 2021.
- FEUERSTEIN, S. **Oracle PL/SQL Programming**. 6th edition. Sebastopol: O'Reilly, 2014.
- FERNANDES, L. **Oracle 9i para desenvolvedores. Oracle Developer 6i. Curso completo**. 1ª edição. Rio de Janeiro: Axcel Books, 2002.
- HEUSER, C. A. **Projeto de Banco de Dados**. 6ª edição. Porto Alegre: Bookman, 2009.
- MACHADO, F. N. R. **Banco de Dados: projeto e implementação**. 4ª edição. São Paulo: Érica, 2020.

---

<sup>8</sup>Eu recomendo a aquisição do livro para quem for trabalhar, no futuro, com bancos de dados. Esse livro é uma ótima fonte de conhecimento e referência.

- COUGO, P. **Modelagem Conceitual e Projeto de Bancos de Dados**. 1ª edição (14ª reimpressão). Rio de Janeiro: Elsevier, 1997.
- DATE, C. J. **Database Design and Relational Theory: normal forms and all that jazz**. 2nd edition. Healdsburg: Apress, 2019.
- DATE, C. J. **Database In Depth: Relational Theory for Practitioners**. 1st ed. Sebastopol: O'Reilly, 2005.
- DATE, C. J. **Relational Theory for Computer Professionals: what relational databases are really all about**. Sebastopol: O'Reilly, 2013.
- DATE, C. J. **SQL and Relational Theory: how to write accurate SQL Code**. 3rd edition. Sebastopol: O'Reilly, 2015.
- DATE, C. J.; DARWEN, H. **Databases, Types, and the Relational Model: the Third Manifesto**. 3rd edition. Massachusetts: Addison Wesley, 2007.
- DATE, C. J. **The New Relational Database Dictionary: terms, concepts, and examples**. 1st edition. Sebastopol: O'Reilly, 2016.
- DATE, C. J.; DARWEN, H.; LORENTZOS, N. A. **Temporal Data And The Relational Model**. 1st edition. San Francisco: Morgan Kaufmann, 2003.
- DATE, C. J.; DARWEN, H.; LORENTZOS, N. A. **Time And Relational Theory: temporal databases in the relational model and SQL**. 1st edition. Waltham: Morgan Kaufmann, 2014.
- DATE, C. J. **The Database Relational Model: a retrospective review and analysis**. Massachusetts: Addison-Wesley, 2001.
- KLEPPMANN, M. **Designing Data-Intensive Applications: the big ideas behind reliable, scalable, and maintainable systems**. 1st edition. Sebastopol: O'Reilly, 2017.
- DATE, C. J. **View Updating & Relational Theory: solving the view update problem**. 1st edition. Sebastopol: O'Reilly, 2013.
- BURLESON, D. K. **Oracle 9i UNIX Administration Handbook**. 1st edition. Berkeley: Osborne, 2002.
- MOTIVALA, B. **Oracle Forms: interactive workbook**. 1st edition. New Jersey: Prentice Hall, 2000.
- SCIORE, E. **Understanding Oracle APEX 20 Application Development**. 3rd edition. Massachusetts: Apress, 2020.
- KYTE, T. **Expert Oracle Database Architecture**. 2nd edition. Healdsburg: Apress, 2010.
- O'Hearn, S. **OCA Oracle Database SQL**. 1st edition. New York: Oracle Press, 2017.

 **Softwares:** para facilitar a preparação do ambiente e evitar o pesadelo inicial de administração de sistemas até que todos os alunos estejam com os softwares instalados e configurados, esta disciplina utilizará uma **máquina virtual**

para o [VirtualBox](#), com todos os softwares, ferramentas, sistemas de bancos de dados e aplicativos instalados. O professor explicará onde obter e como instalar essa máquina virtual;

- ✍ **Material escolar:** você precisará de caderno, lápis (de preferência 6B, lapiseiras são incompatíveis com bancos de dados!), borracha e apontador para fazer anotações nas aulas. Também precisará de algumas folhas de papel almaço A4 com pauta;
- ✂ **Pato de borracha:** é importante você ter um pato de borracha (daqueles amarelinhos para bebês mesmo) para praticar uma técnica chamada de [Rubber Duck Debugging](#)<sup>9</sup> (acredite, funciona!);
- + **Outros:** se necessário, serão detalhados pelo professor.

## 5 Dinâmica da disciplina

Este curso funcionará da seguinte maneira:

1. Você frequentará dois ambientes de aprendizagem: as **aulas presenciais**, nas salas e laboratórios da UVV, e as **aulas online** do curso [CR6.180A: Introdução ao Projeto e Sistemas de Bancos de Dados](#). Tanto as aulas presenciais quanto as aulas online do CR6.180A são **obrigatórias**. Haverá um terceiro ambiente de aprendizagem, o **Oracle Academy**, opcional, para os alunos interessados em aprender mais e receber um certificado oficial da Oracle (altamente recomendado!).
2. As primeiras aulas serão teóricas, para que você tenha uma visão geral de todo o conteúdo da disciplina, de forma relativamente superficial. Depois faremos algumas semanas de aulas práticas, com o uso da máquina virtual, para que você aprenda SQL e como manipular bancos de dados. Essa parte prática consumirá aproximadamente a metade do semestre. Depois voltaremos para as aulas teóricas, aprofundando os conceitos que você estudou e aprendendo conteúdos avançados, incluindo o projeto de bancos de dados, normalização, álgebra relacional e outros assuntos. Durante a segunda metade do semestre as aulas serão teóricas e práticas.
3. A cada semana você tem **leituras obrigatórias** para fazer. Essas leituras estão indicadas no calendário da disciplina e na página “Leituras” do portal [Disciplinas UVV](#).
  - **NÃO ATRASE AS LEITURAS**, o conteúdo de leitura e estudo é muito grande e, se você atrasar, pode não conseguir recuperar o tempo perdido.

---

<sup>9</sup>Ver também: <https://cs50.noticeable.news/posts/rubber-duck-debugging-in-cs-50-ide>.

- Ao final de cada semana de aula haverá, no ambiente do CR6.180A, questionários e exercícios referentes às leituras. Todas essas atividades fazem parte da nota para sua aprovação.
  - O ideal é que você realize as leituras obrigatórias ANTES da aula presencial correspondente na semana, para que você possa discutir e sanar suas dúvidas com o professor.
4. Você também terá **leituras opcionais** a cada semana. Apesar de opcionais, são altamente recomendadas.
  5. **Trabalhos e atividades rápidas:** teremos alguns trabalhos e/ou atividades semanais de fixação da aprendizagem, para casa, com o objetivo de forçar a leitura e o estudo dos capítulos. É uma ótima oportunidade para você verificar se aprendeu a matéria;
  6. **Testes rápidos em sala:** teremos alguns testes rápidos em sala, para verificar se você está mantendo a leitura em dia. Esses testes serão realizados de surpresa, em datas aleatórias, sem aviso prévio. Esses testes são curtos, uma ou duas perguntas discursivas, para serem respondidos em 5 a 10 minutos;
  7. **Vídeos:** além das leituras também estarão listados vídeos obrigatórios ou opcionais para complementar seu aprendizado. O conteúdo dos vídeos obrigatórios fará parte das avaliações.
  8. **Problem Sets (PSETs):** você realizará de 5 a 10 *Problem Sets* nesta disciplina, cada um com uma data específica de entrega. Em geral você terá uma ou duas semanas para responder cada PSET (veja a Seção 5.3 para mais informações);
  9. **Provas:** faremos duas provas presenciais bimestrais, conforme o calendário da UVV. Além dessas provas bimestrais ocorrerão provas e avaliações periódicas no ambiente da CR6.180A.

## 5.1 Leituras, trabalhos e atividades

As atividades semanais de leitura, estudo, trabalhos e atividades rápidas serão a **base de seu aprendizado**. Não espere aprender tudo o que você precisa saber sobre bancos de dados apenas assistindo aulas com o professor: isso não funciona! As aulas têm o objetivo de explicar a matéria mas o aprendizado real só ocorre se você estudar de modo ativo.

A disciplina foi planejada para que você estude, aproximadamente, um capítulo do livro texto de referência por semana, guiado pelas aulas com o professor e pelas tarefas e exercícios. Pode ser difícil nos primeiros dias, até que você se acostume com o ritmo de leitura e estudo necessário, mas depois tudo ficará mais fácil (e você verá os resultados rapidamente!).

Você deve entregar ao professor as respostas dos trabalhos, atividades e exercícios, bem como os resultados obtidos nas atividades práticas. Cada atividade/exercício especificará como essa entrega deverá ser feita e a data limite para a entrega.

## 5.2 Aulas

Tente ler o capítulo indicado para a semana **antes das aulas** daquela semana. Seu aproveitamento será muito melhor pois já terá aprendido uma base e poderá esclarecer dúvidas com o professor. Lembre-se: seu aproveitamento será muito prejudicado se você apenas se limitar a assistir as aulas e não realizar as leituras semanais.

## 5.3 *Problem Sets* (PSETs)

Um *Problem Set* (PSET) é uma lista com um número relativamente pequeno de **questões difíceis**, feitas para testar sua compreensão global do conteúdo estudado durante toda a semana ou um pequeno período de tempo.

Ao contrário das atividades rápidas e trabalhos semanais, que são relativamente fáceis e diretos (pois têm o objetivo de auxiliá-lo no estudo do conteúdo de uma semana), os PSETs são projetados para forçá-lo a pensar profundamente no que você aprendeu durante a semana; para conseguir responder às questões dos PSETs você precisará ter desenvolvido uma compreensão profunda da matéria, muito além do nível de simples “decoreba”.

Deixe-me alertá-lo mais uma vez: os **PSETs são difíceis**, todas as questões são difíceis e algumas são incrivelmente difíceis. Por esse motivo você terá uma ou duas semanas para realizar cada PSET (e não fique chateado se não conseguir responder completamente uma ou mais questões de cada PSet, eles foram feitos para desafiá-lo; mas é importante que, mesmo que você não consiga responder alguma questão, você se esforce e tente ao máximo encontrar a solução).

Para ter uma idéia melhor do que esperar de cada PSET, e porque o uso de PSETs é importante, dê uma olhada no que os alunos do Massachusetts Institute of Technology (MIT) acham do uso de Psets em suas disciplinas:

- *Harbinger of Doom, Despair, and Knowledge: PSETS* (“Arautos da Perdição, Desespero e Conhecimento: PSETS”);
- *A love letter to psets* (“Uma carta de amor aos psets”); e
- *The Process of Psetting* (“O Processo de *Psetting*”).

O grau de dificuldade de cada PSET não atingirá o grau exigido dos alunos do MIT (ainda!), mas mesmo assim serão muito mais difíceis do que você pode estar acostumado (mas afinal... você quer aprender a usar corretamente um sistema de bancos de dados, não é?).

A resolução dos PSETs é **fundamental para seu aprendizado**. Trabalhe diariamente em cada PSET: é melhor fazer várias pequenas partes todos os dias do que tentar fazer tudo de uma vez, de última hora.

Devido ao grau de dificuldade, recomenda-se **trabalhar em pequenos grupos** na resolução de cada PSET, desde que você siga ao pé da letra todas as regras de **integridade acadêmica** do curso (Seção 7), em especial as **políticas sobre trabalho colaborativo** (Seção 7.1).

A questão da integridade acadêmica é muito séria e eu levo isso ao pé da letra. Por favor, não tente entregar o trabalho de um colega (no todo ou até mesmo alguma pequena parte) como seu: existem ferramentas online que verificam plágio em questão de segundos e, além disso, eu realmente leio as respostas de todos os PSets. E pior ainda, você está se enganando...

Para trabalhar presencialmente em pequenos grupos você pode utilizar a biblioteca; se quiser pode marcar grupos online com o uso de ferramentas gratuitas como o [Jitsi](#), o [Google Meet](#) ou o [Zoom](#).

Cada PSET terá instruções específicas que indicam como as respostas devem ser enviadas ao professor e a data limite de entrega.

## 5.4 Monitoria e tutoria

Estão previstas aulas de monitoria<sup>10</sup> para os alunos deste curso mas, neste momento, o monitor está em processo de seleção. Assim que o monitor for conhecido, maiores informações serão dadas.

Durante o semestre o professor realizará algumas poucas seções de tutoria<sup>11</sup>. Essas datas serão divulgadas previamente para o agendamento dos alunos.

# 6 Avaliação da aprendizagem

## 6.1 Provas

Você fará duas provas presenciais durante o semestre, conforme o calendário oficial da UVV (o cronograma da disciplina indica as datas e o conteúdo de cada prova). As provas terão diversos tipos de questões, tais como: discursivas, objetivas, verdadeiro ou falso, interpretação de esquemas e diagramas etc. Note que as provas serão realizadas **sem consulta** e de modo **individual**.

As provas são relativamente extensas mas preparadas para o limite de tempo de uma aula, 01:40h (uma hora e quarenta minutos). É importante que nos dias de prova você não se atrase para que comece a prova no horário marcado. Atenção: depois da primeira pessoa entregar a prova e sair da sala, não será mais permitido a entrada de ninguém.

---

<sup>10</sup>**Monitoria:** uma seção de estudo em grupo (até 20 alunos), sob a orientação de um monitor, com a duração de 1–2 horas. As monitorias expandem o material estudado e introduzem material suplementar não diretamente abordado nas aulas ou nas atividades da semana. Além disso elas são uma chance que você têm de estudar e praticar sob orientação do monitor, sanar dúvidas a respeito do conteúdo, praticar atividades que reforçam o aprendizado e explorar variações dos assuntos discutidos. O monitor pode fazer perguntas sobre a matéria e levar listas de exercícios para serem resolvidas individualmente ou em grupo.

<sup>11</sup>**Tutoria:** uma seção individual para esclarecimento de dúvidas diretamente com o professor, com duração de 15–20 minutos. As tutorias são uma chance de obter ajuda individual, orientações sobre as atividades e os PSETs, e ter o seu progresso de aprendizado verificado (e corrigido, se necessário).

As provas corresponderão a 40% de sua nota final. A **participação nas provas é obrigatória**, ou seja: mesmo que você obtenha nota de aprovação com as outras atividades, se faltar a uma ou mais provas, ficará reprovado na disciplina. Preste atenção ao cronograma da disciplina para não perder nenhuma prova.

## 6.2 PSETs

Você realizará de 5 a 10 PSETs na disciplina, dependendo do tempo e do andamento da turma, e os PSETs corresponderão a 45% de sua nota final na disciplina.

Note que os PSETs, ao contrário das provas, não são atividades obrigatórias, ou seja, se você não entregar algum PSet você ficará com nota 0 (zero) nesse PSET específico, mas não será reprovado por causa disso.

Apesar dos PSETs não serem obrigatórios eles são **fundamentais** para o aprendizado do conteúdo nesta disciplina, conforme já explicado na Seção 5.3, e essa importância é refletida na proporção que os PSETs têm na nota final. Faça todos os PSETs: eles são a base de seu aprendizado!

## 6.3 Trabalhos e atividades

Para auxiliar no aprendizado do conteúdo você fará trabalhos e atividades semanais que corresponderão a 15% de sua nota final.

Da mesma maneira que os PSETs, esses trabalhos e atividades semanais não são obrigatórios, mas importantes para você verificar se aprendeu o conteúdo de cada semana. Se você não entregar algum trabalho ou atividade ficará com nota 0 (zero) nessa atividades específica, mas não será reprovado por causa disso.

## 6.4 Testes rápidos, vídeos etc.

O professor pode, a seu exclusivo critério, indicar outras atividades na disciplina (vídeos, testes rápidos, etc.) que poderão fazer parte de sua nota final ou fazer parte de “pontos extras” para a complementação de notas. Essas atividades, se indicadas, serão detalhadas no tempo oportuno.

## 6.5 Composição da nota final

A nota final da disciplina será composta de:

<b>Avaliação</b>	<b>Proporção na Nota Final</b>
Provas oficiais (3)	40%
PSet (5 a 10)	45%
Trabalhos e atividades	15%

## 7 Integridade acadêmica

Uma questão fundamental relacionada ao trabalho que você fará nesta disciplina (e, na verdade, em qualquer ambiente acadêmico) diz respeito a sua **integridade acadêmica**, ou seja: nós esperamos que suas escolhas sejam íntegras, responsáveis e honestas.

As principais violações acadêmicas que os alunos costumam cometer são: plágio, colaboração não autorizada, “cola” e entregar o trabalho de um colega (total ou parcialmente) como se fosse seu. Por favor, não quebre as regras de integridade acadêmica: você está se enganando, enganando o professor e enganando a UVV.

Para auxiliá-lo a entender melhor o que pode ou o que não pode ser feito, sugiro que você leia o manual *Academic Integrity at MIT: A Handbook for Students*, que é um dos melhores e mais objetivos textos a respeito do assunto:

 [Academic Integrity at MIT: A Handbook for Students](#)

O site com as regras de integridade acadêmica do MIT traz muitas informações resumidas de forma clara e objetiva para os alunos, e também recomendo que você dê uma olhada:

 [Academic Integrity at MIT](#)

Um outro site importante que você deve conhecer é o site com as regras de integridade acadêmica da disciplina *Harvard University: CS50* (a disciplina de introdução à ciência da computação de Harvard). O site da disciplina tem uma lista bem clara de coisas que podem e que não podem ocorrer:

 [Academic Honesty at Harvard CS50](#)

Atenção: **esta disciplina seguirá as regras de integridade acadêmica do MIT e de Harvard**, conforme explicadas nos links acima. É sua responsabilidade ler, e seguir, essas regras. Em caso de dúvidas quanto a alguma regra, entre em contato com o professor. Alguns exemplos do que pode e do que não pode ser feito, baseados nas regras acima citadas de Harvard e do MIT:

### Atitudes permitidas:

- Conversar com os colegas a respeito das atividades, exercícios e PSets, desde que você não esteja pedindo a resposta;
- Discutir os materiais do curso com os colegas, para compreender melhor o conteúdo da matéria e esclarecer dúvidas;
- Ajudar um colega a identificar um erro ou bug em alguma atividade ou exercício, desde que você não faça a correção ou forneça a resposta (o seu colega é que tem que se esforçar, baseado em suas dicas);
- Incorporar algumas linhas de código que você encontrou online na sua própria solução, desde que essas linhas **não sejam a solução** para o problema, e que **cite e identifique** quais foram as linhas (e coloque um link para os originais);

- Estudar as atividades, exercícios e PSets dos semestres passados como forma de aprendizado extra (e não, as atividades e PSets deste semestre não serão iguais aos dos semestres passados);
- Olhar as respostas e códigos de seus colegas para ajudá-los caso eles estejam tendo dificuldades, desde que você apenas identifique onde estão os erros/bugs e não forneça a solução, apenas dê dicas e orientação de como eles podem ser resolvidos;
- Buscar ajuda do professor (incluindo monitores, tutores e alunos mais experientes) para esclarecer a matéria e ajudar na compreensão e resolução das atividades e PSets, desde que você faça uma pergunta específica e não queira a solução;
- Buscar material extra na biblioteca ou na internet para aprender e estudar, desde que esse material extra não seja a solução para as atividades e PSets; e
- Explicar para seu colega como obter a solução de um problema usando frases, diagramas ou pseudocódigo genérico, mas sem mostrar ou fornecer a resposta final.

● **Atitudes proibidas:**

- Pedir a solução de um problema para seu colega, ou procurar na internet uma solução pronta;
- Pedir para ver a solução de um colega para saber se a sua solução está “batendo”;
- Não citar o nome de colegas com os quais você trabalhou em grupo na resolução de alguma atividade ou PSet;
- Não citar a fonte e não indicar o link de linhas de código que você encontrou online e utilizou como parte de sua solução;
- Dar ou mostrar para os colegas a solução para questões que eles ainda não conseguiram responder (você pode ajudar e tirar dúvidas, mas não pode mostrar a sua solução);
- Colar nas provas online (principalmente através da internet), ou seja, durante as provas online não é permitido que os alunos se reúnam de forma online para resolver a prova coletivamente;
- Pegar a resposta de um colega (ou até mesmo pequenas partes da resposta) e apresentar como sua;
- Pagar para alguém fazer o trabalho em seu lugar;
- Postar as questões em fóruns e sites na internet, solicitando que alguém as responda;
- Dividir com os colegas as atividades e exercícios de forma que cada um faça uma pequena parte do trabalho e depois compartilhe com o restante (você deve trabalhar em todos os problemas);

- Pegar o trabalho de um colega, mudar algumas frases e vírgulas, e entregar como se fosse seu; e
- Qualquer outra coisa que sua consciência considere desonesta.

## 7.1 Política sobre trabalho colaborativo

Cientistas, programadores, estatísticos, engenheiros e, em geral, todas as pessoas ligadas à área de “STEM” (*Science, Technology, Engineering, Math*) costumam trabalhar em grupos. As interações sociais são críticas para o resultado de seus estudos e grandes idéias costumam aparecer em discussões com os colegas. Assim, nesta disciplina, o trabalho em grupo é fortemente recomendado e estimulado, desde que algumas regras sejam seguidas.

Seguiremos integralmente a política sobre trabalho colaborativo adotada na disciplina “*MIT 6.001: Structure and Interpretation of Computer Programs*” que, em tradução livre, é a seguinte<sup>12</sup>:

“A maioria das pessoas aprende com mais eficácia quando estuda em pequenos grupos e coopera de várias outras maneiras no dever de casa. Isso pode ser particularmente verdadeiro em tarefas que envolvem programação, onde trabalhar com um parceiro geralmente ajuda a evitar erros e descuidos. Somos totalmente a favor desse tipo de cooperação, desde que todos os participantes se envolvam ativamente em todos os aspectos do trabalho — não apenas dividam a tarefa e cada um fica responsável apenas por uma parte.

[...] encorajamos você a trabalhar com uma ou duas outras pessoas mas, ao entregar seu projeto, você deve identificar com quem trabalhou. Esperamos, no entanto, que você esteja envolvido em todos os aspectos do projeto, que escreva e comente seu próprio conjunto de código e que escreva seus resultados separadamente. Quando você entrega um material com seu nome, presumimos que você está certificando que este é o seu trabalho e que esteve envolvido em todos os aspectos dele. Não entregue apenas uma cópia de um único arquivo; escreva sua própria versão. Isso significa que você mesmo cria esse arquivo e não apenas anota uma cópia que recebeu de outra pessoa. Sabemos que isso pode soar como replicação de trabalho, mas uma parte importante do aprendizado do material é tornar o processo ativo, especialmente com relação à edição, execução e depuração de software, o que você faz garantindo que criou e consegue explicar sua solução.

Aqui está um exemplo de como um bom trabalho de cooperação deve ocorrer:

- Ambos (todos) sentam-se com lápis e papel e planejam juntos como resolverão e testarão as coisas. Você vai com seus colegas para

---

<sup>12</sup>O documento original está [disponível na internet](#).

algum local adequado e sentam-se em máquinas adjacentes. Após você resolver um problema, você verifica se seus colegas também conseguiram e se estão todos no mesmo ritmo. Quando um de vocês tem um problema, os outros olham e tentam ajudar. Por exemplo, seu parceiro tem um bug em uma parte e você pode apontar onde está o bug e como corrigi-lo. Em cada parte do problema, você escreve seu próprio código e solução, buscando ajuda dos outros quando tem dificuldades. No trabalho final, cada um de vocês lista os nomes de todos os seus colaboradores.

Aqui está um exemplo de cooperação inapropriada:

- Você envia ao seu amigo uma cópia do seu trabalho até agora. Ele continua o seu trabalho para concluir o procedimento que você não conseguiu, e corrige um bug ou erro em outra parte. Cada um de vocês envia este código e solução compartilhados. Mesmo que você liste os nomes um do outro como colaboradores, esta é uma colaboração inadequada porque vocês não estiveram envolvidos em todos os aspectos do trabalho — cada um de vocês não escreveu sua própria implementação, mesmo que para um plano comum, e você compartilhou um conjunto comum de código e redação.

Não listar o nome de um colaborador será considerado trapaça. Da mesma forma, lembre-se de que copiar o trabalho de outra pessoa e apresentá-lo como seu próprio trabalho é uma ofensa acadêmica grave e será tratado como tal.

Em geral, recomendamos fortemente que você trabalhe em grupo. É uma maneira muito eficaz de detectar erros conceituais e outros, e de refinar o pensamento e a compreensão [...].”

Em resumo: colabore com seus colegas, estude em grupo, mas siga as regras de trabalho colaborativo. Uma dica: antes de participar de um grupo de estudo, tente resolver as questões sozinho primeiro; isso lhe dará uma visão geral dos problemas a serem resolvidos, do grau de dificuldade, e proporcionará que você vislumbre um caminho para a solução que pode ser discutido com seus colegas.

## 7.2 Cláusula de arrependimento

Esta disciplina oferecerá uma **cláusula de arrependimento** conforme praticada no curso *Harvard CS50*<sup>13</sup>: se você cometer algum ato que viole as políticas de integridade acadêmica aqui estabelecidas mas, em até 72 horas, se arrepender e comunicar o fato ao professor, poderá ter a nota reduzida ou zerada nesse trabalho/PSet específico, mas o caso não será levado à coordenação.

---

<sup>13</sup>Ver *Teaching Academic Honesty in CS50*, de David Malan, Brian Yu e Doug Lloyd.

## 8 Perguntas frequentes

- **Sou obrigado a fazer as atividades e trabalhos semanais?** Depende. Algumas podem ser opcionais e, se esse for o caso, você não é obrigado (embora elas podem contribuir para seu aprendizado). Se as atividades em questão contribuírem para a nota final da disciplina, elas são obrigatórias (se você não fizer, ficará com nota zero nessa atividade).
- **Sou obrigado a fazer os PSETs?** Bem, considerando que os PSETs correspondem, em conjunto, à 45% de sua nota de aprovação, eles são obrigatórios. Se você não entregar um PSET ficará com nota zero nesse PSET.
- **Sou obrigado a fazer as provas?** Sim. Além de corresponderem a 40% de sua nota final, se você falta a uma ou mais provas poderá ficar reprovado.
- **Onde os avisos sobre a disciplina serão postados?** No site [Disciplinas UVV](#) e no Portal do Aluno. Visite esses sites com frequência para ficar atualizado com as últimas notícias.
- **Por que tenho que seguir as regras de integridade acadêmica do MIT e de Harvard?** Em primeiro lugar porque você está na graduação e tem a obrigação de agir com honestidade acadêmica. Em segundo lugar porque as regras do MIT e de Harvard são claras, objetivas e disponíveis livremente na internet para que qualquer pessoa possa utilizar. E, para saber se os alunos estão lendo este documento, a primeira pessoa de cada turma que seguir todas as instruções e links de referências detalhadas aqui e achar e me mandar um e-mail com o texto da “primeira lei de abramantes” ganhará uma caixa de bombom. E, por último, eu (seu professor) acredito em integridade acadêmica e acredito que essas regras são importantes.
- **As normas de integridade acadêmica serão obrigatórias?** Sim, serão. Por favor, não tente a sorte copiando o trabalho de seu colega ou, se assim o fizer, utilize a cláusula de arrependimento. Isso evitará maiores problemas e dissabores para todo mundo.
- **Li o *syllabus* mas ainda tenho dúvidas!** Entre em contato com o professor!