

Apresentação

Professor, disciplina e sistema de avaliação

Prof. Me. Fausto G. Cintra



❖ Versão 4.6, casado, 2 filhas (9 e 13 anos)

❖ Formação acadêmica

- Mestre Interdisciplinar em Desenvolvimento Regional (Uni-FACEF, 2014)
- Bacharel em Direito (Unesp Franca, 2008)
- Bacharel em Ciência da Computação com ênfase em Análise de Sistemas (Unifran, 1997)

❖ Experiência profissional

- Andes Informática: programador (1997-2001)
- Prefeitura Municipal de Franca: analista de sistemas (2001-2005)
- Centro Universitário Municipal de Franca (Uni-FACEF): oficial de tecnologia e informática (2005-2014)
- Universidade de Franca: docente (2013-2014)
- Fatec Franca: docente (2014-)
- Uni-FACEF: docente (2022-)



E quem é você?

- ❖ Eu também gostaria de saber um pouco mais sobre você. Convido-o(a) a preencher um formulário no seguinte link:



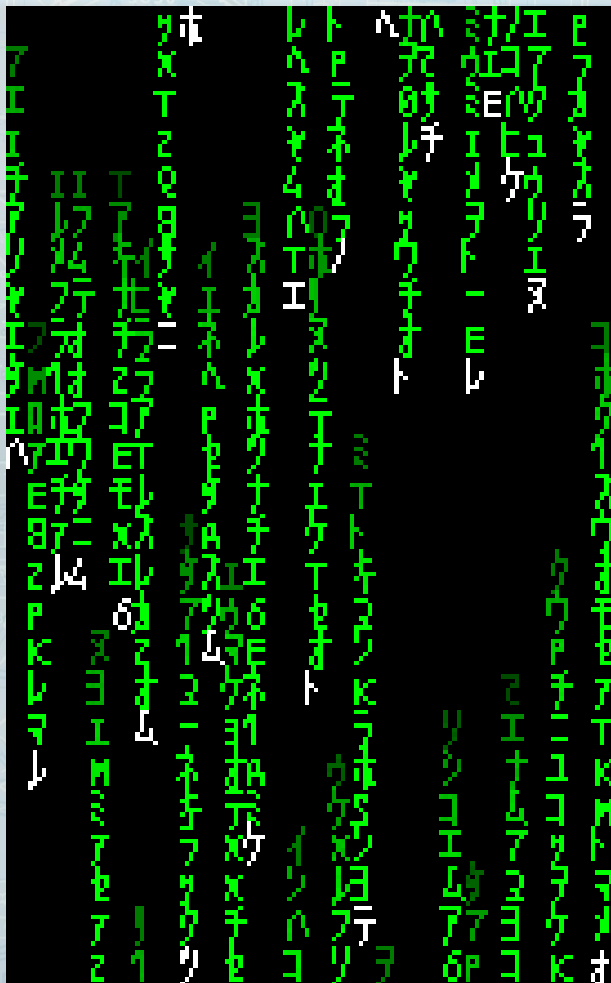
tiny.cc/apresenta2023

- ❖ Responder é rápido, leva menos de 3 minutos, e permite ao seu professor conhecê-lo(a) um pouco melhor para ajudar nos seus estudos

PARA QUÊ SERVEM OS COMPUTADORES?



Você já se perguntou alguma vez...



- ❖ Por que é que os computadores foram inventados?
- ❖ Que tipos de problemas eles são capazes de resolver?



`for(let i = 0; i < 1000; i++)`



❖ Computadores sabem fazer apenas duas coisas:

- repetir; e
- contar

- ❖ Mas eles fazem essas duas coisas de forma mais rápida e mais eficiente que qualquer ser humano
- ❖ Tudo o que fazemos no computador é uma consequência ou variação dessas duas operações



Mas eles não sabem fazer isso sozinhos...

(não totalmente, pelo menos por enquanto...)



- ❖ Ainda cabe ao ser humano instruir o computador a fazer as **repetições** e **contagens** de modo a que a máquina seja, de fato, útil
- ❖ As instruções são dadas aos computadores por meios de **ALGORITMOS** representados por meio de **LINGUAGENS DE PROGRAMAÇÃO**.



A responsabilidade também é sua



- ❖ Partindo do princípio de que o computador sempre fará aquilo que lhe foi ordenado pelo algoritmo, **o processo de computação será tão eficiente quanto o próprio algoritmo**
- ❖ Outro fator a considerar é que o **computador é uma máquina de recursos finitos**, isto é, a capacidade de seu processador, da memória e das unidades de armazenamento tem limites, dentro dos quais os algoritmos devem trabalhar
- ❖ Em suma: **a responsabilidade pela eficiência do computador cabe, em grande medida, ao programador do algoritmo**



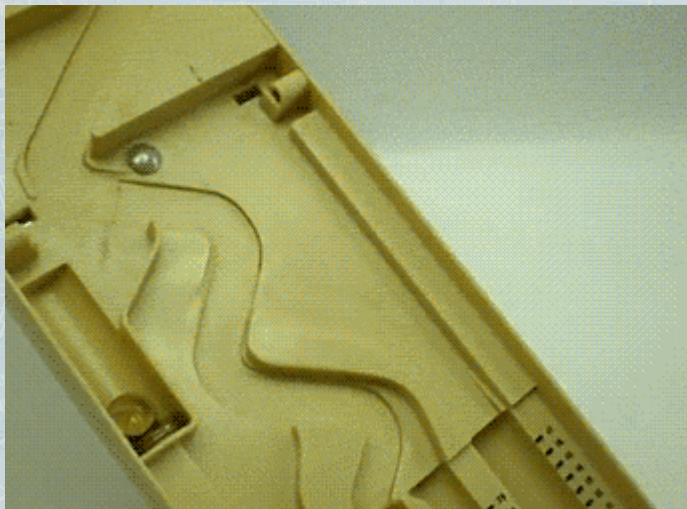
Estratégia, do grego *στρατηγία*



- ❖ Quase sempre, há mais de uma forma de resolver o mesmo problema, ou seja, diferentes algoritmos
- ❖ Naturalmente, entre esses algoritmos, alguns utilizarão melhor os recursos do computador do que outros, e serão considerados mais eficientes
- ❖ É preciso elaborar estratégias para extrair o melhor do computador



Aplicações práticas de contagem e repetição



- ❖ Entre as aplicações que melhor tiram proveito das operações de contagem e repetição feitas por computadores, estão:
 - A organização de uma lista de valores segundo um critério específico → **ALGORITMOS DE ORDENAÇÃO**
 - A procura de um valor em uma lista de valores previamente organizada → **ALGORITMOS DE BUSCA**
- ❖ A vantagem dos computadores sobre o trabalho humano, nesses casos, cresce à medida que o volume de dados a serem processados aumenta
- ❖ As ineficiências dos algoritmos também ficam mais evidentes na mesma proporção do crescimento do volume de dados

ALGORITMOS DIFERENTES, RESULTADOS IGUAIS, EFICIÊNCIA DISTINTA



Um exemplo simples

Suponha que precisemos encontrar um valor, digamos 78, na lista ordenada abaixo.

4	11	23	29	37	41	54	60	67	72	78	85	96
---	----	----	----	----	----	----	----	----	----	----	----	----

A estratégia mais elementar é partir do início e comparar o valor de cada posição ao valor que buscamos, até encontrá-lo.

4	11	23	29	37	41	54	60	67	72	78	85	96
---	----	----	----	----	----	----	----	----	----	----	----	----

11 comparações

Fica evidente que, quanto mais ao final o valor buscado estiver, mais comparações serão necessárias para achá-lo. E isso se torna um problema cada vez maior no mesmo passo que o número de elementos da lista aumenta.

Uma estratégia melhor (1)

1. Vamos encontrar o elemento que está no meio da lista

Meio da lista

4	11	23	29	37	41	54	60	67	72	78	85	96
---	----	----	----	----	----	----	----	----	----	----	----	----

2. O valor que buscamos (78) é NÃO É IGUAL (**1ª comparação**) ao valor do que aquele que está no meio da lista, mas é MAIOR (**2ª comparação**) que ele. Assim, podemos descartar a primeira metade da lista (e também o valor do meio).

4	11	23	29	37	41	54	60	67	72	78	85	96
---	----	----	----	----	----	----	----	----	----	----	----	----

3. Repetimos o processo de encontrar o meio na sublista que restou. Como o número de elementos é par, fazemos um arredondamento para baixo

Meio da sublista

4	11	23	29	37	41	54	60	67	72	78	85	96
---	----	----	----	----	----	----	----	----	----	----	----	----

Uma estratégia melhor (2)

4. O elemento que buscamos (78) é NÃO É IGUAL (**3ª comparação**) ao novo valor central, sendo MAIOR (**4ª comparação**) que ele, de modo que podemos descartá-lo juntamente com a primeira metade da sublista.

4	11	23	29	37	41	54	60	67	72	78	85	96
---	----	----	----	----	----	----	----	----	----	----	----	----

5. Encontramos novamente o elemento médio entre os valores que restaram.

Meio da sublista

4	11	23	29	37	41	54	60	67	72	78	85	96
---	----	----	----	----	----	----	----	----	----	----	----	----

6. O valor de busca (78) NÃO É IGUAL (**5ª comparação**), mas é MENOR (**6ª comparação**) que o valor do meio que consideramos agora. Podemos descartar, então, o valor médio e a segunda metade da lista.

4	11	23	29	37	41	54	60	67	72	78	85	96
---	----	----	----	----	----	----	----	----	----	----	----	----

Uma estratégia melhor (3)

7. O valor que resta passa a ser considerado o meio dos valores que restaram. Por fim, concluímos que o valor de busca (78) é IGUAL (**7ª comparação**) ao valor médio, terminando nossa busca.

Meio da sublista

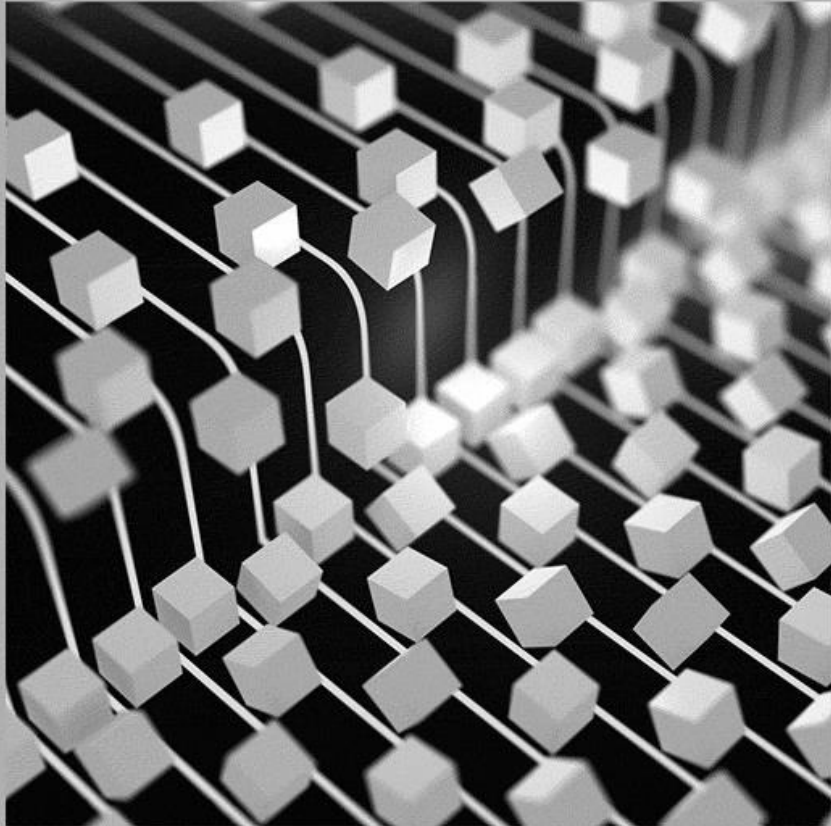
4	11	23	29	37	41	54	60	67	72	78	85	96
---	----	----	----	----	----	----	----	----	----	----	----	----

Portanto, com essa estratégia, usamos **sete comparações** contra as **onze comparações** da primeira tentativa. A diferença pode parecer pouca, mas aumenta conforme o número de elementos na lista cresce, como teremos a oportunidade de demonstrar durante as aulas da disciplina.

A ORGANIZAÇÃO DOS DADOS INFLUENCIA A EFICIÊNCIA DOS ALGORITMOS



Eficiência e organização andam unidas



- ❖ No exemplo de busca que vimos, o algoritmo que utilizou a estratégia mais eficiente só funciona como esperado se os valores estiverem previamente ordenados
- ❖ Em outras palavras, os valores estavam dispostos em uma ESTRUTURA DE DADOS chamada **lista linear ordenada**
- ❖ Dessa situação, é possível concluir que os algoritmos para a resolução de problemas computacionais podem exigir que os dados estejam organizados de uma forma específica para que a máxima eficiência seja alcançada



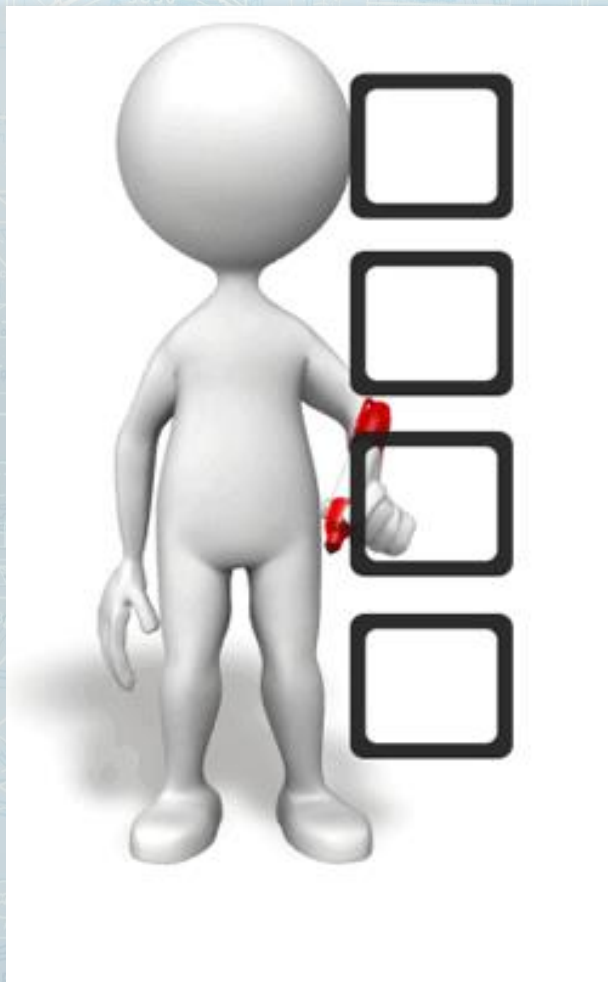
Tipos de Abstratos de Dados



- ❖ Do ponto de vista do programador, muitas vezes é conveniente pensar nas estruturas de dados em termos das operações que elas suportam (algoritmos), e não da maneira como elas são implementadas
 - Imagine alguém que tirou a habilitação em um Fusca. Essa pessoa será capaz de dirigir qualquer outro automóvel que tenha volante, câmbio e pedais dispostos da mesma forma como em um Fusca, mesmo que o motor seja completamente diferente
- ❖ Uma estrutura de dados definida dessa forma é chamada de um Tipo Abstrato de Dados (TAD)

ESTRUTURAS DE DADOS: A DISCIPLINA

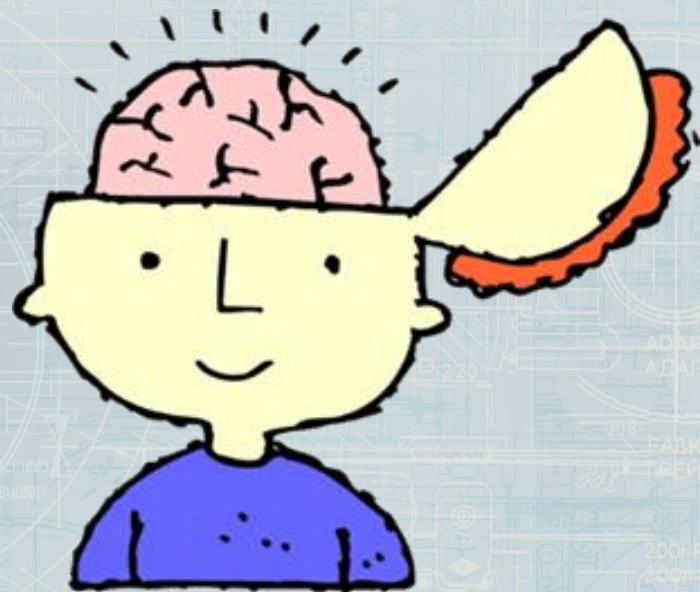
Objetivos de aprendizagem



- ❖ Entender e criar algoritmos de nível não-elementar
- ❖ Compreender e utilizar estruturas de dados lineares na resolução de problemas
- ❖ Compreender e simular o funcionamento de algoritmos de ordenação
- ❖ Entender e criar aplicações de busca sequencial e busca binária
- ❖ Utilizar as técnicas de resolução de problemas no desenvolvimento de programas



Vamos utilizar:



- ❖ A linguagem Python
 - Mas: o raciocínio é mais importante que a linguagem de programação!
- ❖ Um repositório no [GitHub](#)
- ❖ **Um caderno** (importante!)
- ❖ Livros e recursos da Internet
- ❖ E, principalmente, a cabeça!



Como serão as aulas?



- ❖ As aulas alternarão a parte teórico-expositiva e parte prática com exercícios, usando o Visual Studio Code e o Python
- ❖ O uso do caderno é importante para colher anotações



O assunto aqui é sério



- ❖ **Estruturas de Dados é uma das disciplinas mais técnicas** de qualquer curso superior da área de computação
- ❖ Ela representa um **desafio acentuado** para a maioria dos estudantes. **Não é raro que aqueles que não tenham se dedicado o suficiente fiquem de dependência**
- ❖ Portanto, recomenda-se atenção redobrada e, principalmente, **não deixar passar uma dúvida sequer** sem perguntar ao professor



REFERÊNCIAS BÁSICAS



- ❖ EDELWEISS, Nina; GALANTE, Renata. **Estruturas de dados.** Porto Alegre, Bookman, 2009
- ❖ Disponível na Biblioteca da Fatec Franca



- ❖ PEREIRA, Silvio do Lago.
Estruturas de dados fundamentais: conceitos e aplicações. 12. ed. rev. e atual. São Paulo: Érica, 2008
- ❖ Disponível na Biblioteca da Fatec Franca



PYTHON E ORIENTAÇÃO A OBJETOS

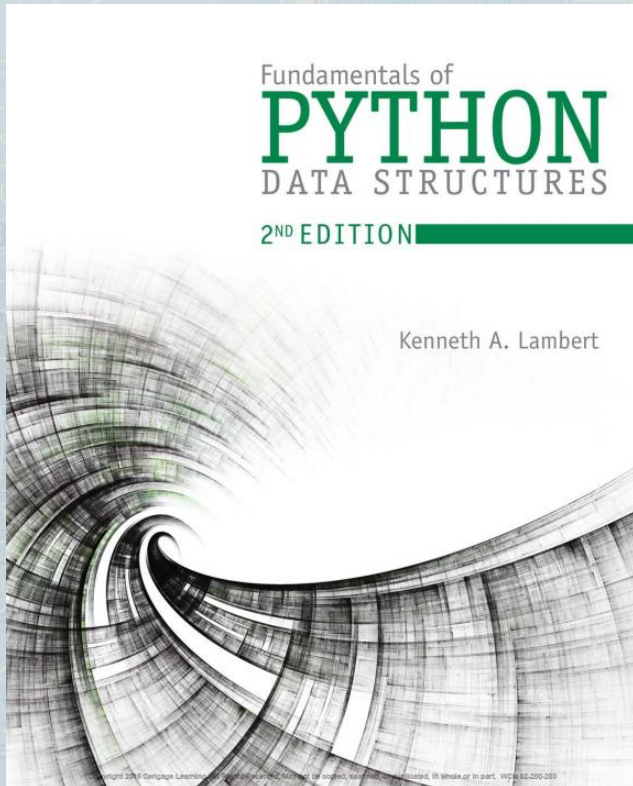
Curso PY-14



Caelum:

❖ CAELUM. Python e orientação a objetos.

- ❖ Disponível na pasta Material de Aulas do canal Geral na equipe do Microsoft Teams



❖ LAMBERT, Kenneth A. **Fundamentals of Python Data Structures.** Cengage, 2019.

❖ <https://b-ok.lat/book/5275575/e1bc68>



❖ MASANORI, Fernando. **Python para Zumbis.**



❖ <https://www.youtube.com/playlist?list=PLUukMN0DTKCTbzhbYe2jdF4cr8MOWClXc>
(playlist)

SISTEMA DE AVALIAÇÃO



Visão geral



- ❖ O sistema de avaliação da disciplina será constituído por:
 - Duas provas (P1 e P2), cada qual valendo 10,0 (dez) pontos, com pesos diferentes
 - Dois trabalhos (T1 e T2), valendo 10,0 (dez) pontos, pesos 1 e 2, respectivamente
 - Participação e assiduidade (PA), avaliadas a critério do professor, valendo até 1,0 (um) ponto, **adicionada diretamente à médias das avaliações anteriores**
- ❖ A média final (MF) será dada pela fórmula

$$MF = \frac{(P1 * 3) + T1 + (P2 * 4) + (T2 * 2)}{10} + PA$$



Provas P1 e P2



- ❖ Cada uma vale 10,0 pontos, pesos 3 e 4
- ❖ Consistirão em exercícios práticos de identificação e utilização de algoritmos e estruturas de dados
- ❖ Serão aplicadas nas datas especificadas na tabela ao final desta apresentação
 - Prova P1: matéria até a semana anterior à aplicação
 - Prova P2: matéria entre a P1 e a matéria até a semana anterior à aplicação
- ❖ **Alunos que, por qualquer motivo, não puderem prestar a prova na data marcada, farão uma nova diferente, em data a ser definida pelo professor**



Trabalhos T1 e T2



- ❖ Valerão 10,0 (dez) pontos, pesos 1 e 2, respectivamente
- ❖ Consistirão na elaboração e entrega de trabalhos de cunho teórico ou prático sobre o conteúdo da disciplina
- ❖ Deverão ser entregues até as datas especificadas na tabela ao final da apresentação, exclusivamente pelo Microsoft Teams
- ❖ **IMPORTANTE: não haverá substituição para a nota dos trabalhos T1 e T2!**



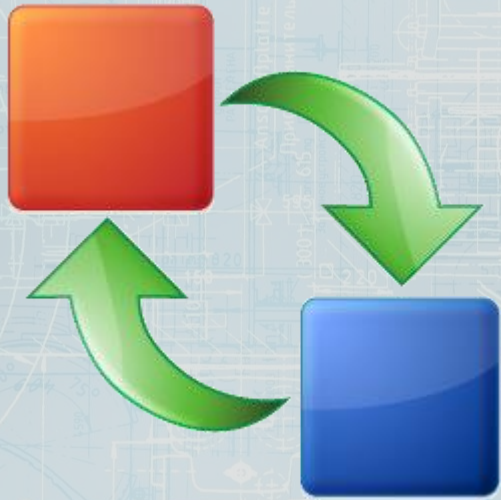
Participação e assiduidade (PA)



- ❖ A participação ativa do aluno durante as aulas, bem como a assiduidade será valorizada e recompensada com até 1,0 ponto, a critério do professor
 - As atividades que forem propostas durante algumas aulas, com entrega pelo Microsoft Teams, serão contabilizadas na nota de participação
- ❖ Questione, duvide, peça para repetir a explicação, mostre outra solução –
PARTICIPE ATIVAMENTE DA AULA!



Prova Substitutiva (PS)



- ❖ Terá direito à PS o aluno que não tiver obtido a média mínima 6,0 (seis) após o cálculo da média final das avaliações regulares
 - Abrangerá todo o conteúdo ministrado na disciplina
- ❖ Valerá 10,0, no mesmo formato das provas P1 e P2. A nota obtida na PS substituirá a nota P1 ou a P2, a que for menor
- ❖ **Não haverá a possibilidade de usar a nota da PS para substituir notas de trabalho (T1 e T2)**



Dicas importantes



- ❖ É de extrema importância que o aluno faça as atividades propostas e as entregue na data correta
- ❖ Nota perdida com trabalhos não entregues ou malfeitos **NÃO É RECUPERÁVEL** com Prova Substitutiva
- ❖ Sem a nota do trabalho, participação e assiduidade, o aluno somente será aprovado caso gabarite as duas provas (P1 e P2), o que é bastante improvável



Problemas?



- ❖ Na vida, é comum fazermos um planejamento e não conseguir executá-lo por motivos alheios à nossa vontade:
 - Trabalho (ou a falta dele)
 - Doenças
 - ...
- ❖ Quero ser seu parceiro nessa jornada. Não interrompa seus estudos antes de conversar comigo ou com os demais professores. Sempre há uma forma de podermos ajudar!
- ❖ **Não se esqueça de se apresentar respondendo ao formulário em tiny.cc/apresenta2023**



Obrigado!



Estou sempre à disposição, seja pelo
chat do Teams ou pelo e-mail

professor@faustocintra.com.br



Calendário de Atividades* – noturno

Atividade	Data ou prazo final
Instruções do T1	Até 24/03 (sex)
Entrega do T1	09/04 (dom), até 23h55
P1	13 e 14/04 (qui/sex)
Instruções do T2	Até 26/05 (sex)
Entrega do T2	11/06 (dom), até 23h55
P2	15 e 16/06 (qui/sex)
PS	23/06 (sex)
* Datas previstas. Sujeitas a alterações circunstanciais	