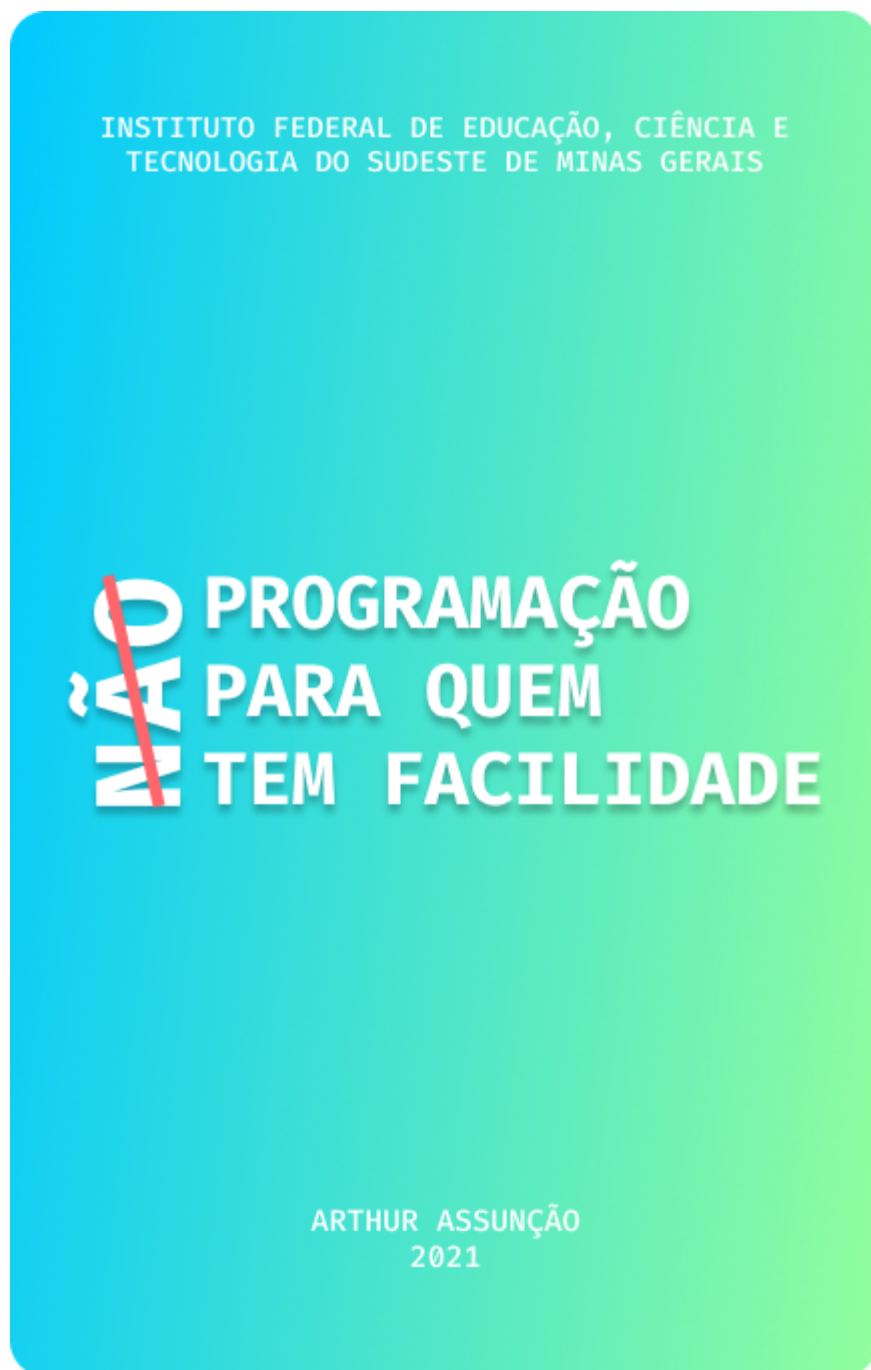


Ebook - Programação para quem não tem facilidade - Capítulo 0: Conhecendo a metodologia



1ª Edição

Material desenvolvido para o ensino de Lógica de Programação e Programação 1.
E-book (2021)

Esta obra pode ser reproduzida, no todo ou em parte, desde que citada a fonte.

ASSUNÇÃO, Arthur Nascimento. Programação para quem não tem facilidade. 1ª ed. Ouro Preto. 2021.

E-book, no formato PDF (indefinido p.)

1. Programação. 2. Lógica. 3. Informática. 4. Aprendizado. Assunção, Arthur Nascimento.



Ebook criado por Arthur Assuncao, professor do IFSudesteMG e desenvolvedor Front-end.

Este ebook pode ser acesso pela url <https://www.notion.so/Conceitos-b-sicos-de-inform-tica-d356a28d68af42b5929eb05011bc8168>.

Sobre o livro e os 7 pilares

Introdução

Programação de computadores ou somente programação é costuma ser um conteúdo ou disciplina muito difícil, com **alta taxa de reprovações**, tem diversos artigos que apresentam isso, mas não há porque apresentarmos aqui. Contudo, como professor e aluno, ocasionalmente, vejo muitos alunos **falharem no aprendizado de programação**, metade da turma costuma não entender nada, uns 5 em 30 costumam ter facilidade, ou seja, 25 em 30 alunos costumam ter alguma ou muita dificuldade.

"Arthur, Mas por que isso acontece?"

Programação envolve **conhecimentos novos**, que a maioria de nós nunca vimos, e precisa de uma lógica (**raciocínio lógico**) muito bem desenvolvida, algo que nossas escolas não conseguem nos ajudar a desenvolver, além disso, muitos professores continuam ensinando de forma totalmente tradicional, sem tentar explicar de novas formas com novos métodos. Por fim, programação não tem decoreba ou cópia, não tem uma receita de bolo como muitas disciplinas, sempre é necessário resolver um problema com conhecimentos prévios e, então, passar para o código.

Não entendeu ainda? Explico com exemplos. Em história podemos aprender decorando como foi a história, assim na prova podemos apenas responder o que decoramos e pronto. Claro que se você entender o contexto, as ideias por trás da época e situação, saberá muito mais, mas basta decorar para aprender um mínimo e conseguir o mínimo 6 nas provas. Em matemática você tem mais dificuldade, mas

geralmente fazendo diversos exercícios, você chega ao ponto que o exercício se torna uma repetição do que você já fez, basta trocar valores, contudo ainda assim matemática é uma das disciplinas que nossos alunos tem mais dificuldade. A programação se aproxima disso, porém é ainda mais complexa, pois **envolve conceitos novos e a resolução do problema** necessita de mais conhecimentos. Pense na programação como uma questão de ENEM onde há diversos temas interligados e você precisa usar as ferramentas que tem para resolvê-lo, não é a toa que a maioria dos alunos não se da bem com o ENEM.

Então como posso ajudar você a aprender programação? Qual é esse método didático? "Arraste para cima e se inscreva na semana da programação que te explico", brincadeira. Vou explicar o que me motivou a montar esse livro e depois eu explicarei os pilares da metodologia.

Desde que comecei a ensinar programação em 2019, tive dificuldades para ensinar alguns alunos, tentei algumas formas como usar **Scratch, Portugol** (aquele bonito do portugol studio) e até explicações com **fluxogramas**. Todas essas formas são positivas, porém faltava serem sistemáticas, seguirem uma ideia clara e sistemática. Em 2020 e 2021, ajudei alunos de cursos de instituições públicas, inclusive universidades públicas, alguns eram alunos bastante motivados, tentavam e queriam entender, mas tinham algumas dúvidas e eu costumo não dar nenhuma resposta pronta, então sempre os fiz pensar sobre suas ações e vi algumas boas evoluções. Contudo, imagino que a maior parte da turma desses alunos, com certeza, estava com as mesmas dúvidas, também estava com dificuldade para aprender. Por conta disso, entra a minha ideia de ensino de programação, não é nada novo ou inovador, mas é algo que ninguém costuma fazer e muito menos fazer de forma **sistemática**.

Público-alvo

Se você reparar bem, este livro é para qualquer aluno de programação, porém ele é **especialmente dirigido aos alunos que tem alguma dificuldade** ou, em outras palavras, ele é menos recomendado aos alunos que tem facilidade com programação, esses poderão considerar o livro repetitivo ou cansativo, pois para eles qualquer livro serve. Então, se você não faz parte dessa nata que tem facilidade, seja bem-vindo e desfrute das metodologias utilizadas pelo livro.

Pilares do método

Os pilares do método são simples e estão no meio de qualquer ensino, mas lembro que a ideia é fazer isso de forma sistemática. Abaixo apresento uma frase sobre o

que é ser um programador e desenvolver códigos.

Desenvolver códigos envolve resolver problemas, assim você primeiro entende o problema, depois divide o problema em pequenas partes e tenta resolver cada parte passo a passo, se necessário pesquisado no Google ou fórum de desenvolvedores.

Assim, os pilares são ilustrados abaixo e devem ser apresentados após uma breve explicação ao conteúdo abordado:

1. Exemplos

Quando mais exemplos temos, mais bagagem teremos para novos problemas, pois boa parte dos problemas são resolvidos com soluções ou parte de soluções de outros problemas, portanto quanto mais diversos forem os exemplos, mais fácil será, afinal os exercícios ou momentos práticos deixam de ser totalmente novos e passam a ser apenas parcialmente novos. Por isso, ver muitos exemplos é excelente, pois te ajuda ao conhecer um novo conteúdo. Este pilar será conhecido como **Veja exemplos**.

2. Aprender a partir de erros

Quando você erra, você aprende como não fazer, ao errar algumas vezes você aprende qual a forma correta de se fazer. Além disso, o erro é mais forte que o acerto para forçar a gravação de uma informação, não tenho comprovações científicas para essa afirmação, contudo pense e me diga, *quantas vezes você errou e nunca mais errou aquilo, porque sempre se lembrava do erro e então fazia o correto?* Ainda há outro motivo para essa abordagem, todos nós vamos nos deparar com erros, se eu lhe mostro alguns erros no momento do aprendizado, você estará preparado para lidar com tais erros. Seria como se eu te treinasse em um labirinto antes de te soltar nesse labirinto, você já sabe quais as soluções para os principais erros, então quando se deparar com um deles, saberá o que fazer. Este pilar será conhecido como **Aprenda com erros!**.

3. Perguntas para reflexão

Eu posso te passar um pedaço de código e dizer: *"este código verifica se a idade é maior que 18 e, se for, imprime uma mensagem dizendo que o usuário é maior de idade"*. Porém, este tipo de abordagem não funciona com todos, principalmente na

programação. O ideal é eu lhe passar um código e perguntar: *"me diga o que esse código faz"*. Então você entenderá com os seus próprios olhos e mente, será um conceito mais bem fixado que apenas decorar que o código x faz algo. Agora imagine que eu peça para você alterar, a cada momento, uma parte do código e me dizer o que acontece, você irá entender muito melhor o que as suas ações e alterações estão causando no código. Este pilar será conhecido como **Refleta!**.

4. Passo a passo

Programar é criar algoritmos em linguagens de programação, algoritmos são sequências de passos finitos para fazer alguma tarefa. Muita gente tem dificuldade em detalhar o passo a passo para realizar alguma tarefa porque não tem o costume de fazer isso, então programar se torna mais difícil, pois as pessoas têm dificuldade na criação de passo a passo e para organizar ideias em suas "cabeças". Portanto, ao ensinar o passo a passo na montagem de código, ensinamos o aluno a pensar como programador. Este pilar será conhecido como **Passo a passo**.

5. Dividir em subproblemas

Na computação existe um conceito chamado **dividir para conquistar**, este conceito envolve a divisão de problemas em problemas menores para facilitar a solução. Pense comigo: você precisa montar uma cómoda, como você irá realizar essa tarefa? Como dividiria o problema em problemas menores? Te dou uma resposta possível. Você poderia separar nos seguintes sub-problemas:

1. Pegar o manual;
2. Juntar os pregos e parafusos e os manter em um local a vista;
3. Juntar maçanetas e dobradiças e as manter em um local a vista;
4. Juntar peças semelhantes, como portas de gaveta com porta de gaveta.
5. Montar a estrutura das gavetas;
6. Colocar as maçanetas nas gavetas
7. Montar a estrutura da cómoda;
8. Colocar as dobradiças e outros itens na estrutura da cómoda;
9. Colocar as gavetas na cómoda;

Perceba que é uma espécie de algoritmo, porém cada parte é uma tarefa diferente e sem muitos detalhes, logo uma subtarefa ou subproblema para ser resolvido.

Pensar em dividir em subproblemas é essencial para qualquer pessoa, pois facilita nossa resolução de problemas na vida.



Não confuda dividir em subproblemas com passo a passo, o passo a passo é a montagem do algoritmo, com detalhes, já dividir para conquistar é separar o algoritmo em partes menores para facilitar o entendimento e desenvolvimento da solução.

Este pilar será conhecido como **Dividir para conquistar**.

6. Leitura de código

É fundamental que o programador saiba ler os códigos, saiba o que cada comando faz para realizar testes de mesa com papel ou mentalmente, assim o aluno não se torna um programador que trabalha na tentativa e erro, ele lê o código e avalia como funciona e o que deve ser corrigido. Este pilar será conhecido como **Leia o código!**.

7. Exercícios

Quando mais exercícios forem realizados mais fácil será fixar um conteúdo, não é a toa que falamos que se aprende matemática resolvendo exercícios, com a computação a prática, também, leva a perfeição.

Limitações do ensino de programação

Pedagogos e outros profissionais costumam dizer que o melhor é integrarmos conteúdos, ou seja, incluir um conteúdo em outros ou mesmo levar o conteúdo para a realidade do aluno, porém nem sempre isso é possível, muitos exercícios são cansativos e parecem inúteis, contudo é uma realidade, não é possível ter todos exercícios legais e bacanas no início da programação.

Sobre os autores

Prazer, sou o **Arthur Assuncao**, professor de computação, formado em Sistemas para Internet no IFSudesteMG campus Barbacena e com mestrado em computação na UFOP (Universidade Federal de Ouro Preto). Gosto muito de estudar, de aprender e de cuidar da própria saúde, por isso me sinto bastante realizado por ter a possibilidade de investir nessas áreas da minha vida. Por conta disso, eu sei

muita coisa em relação a desenvolvimento de software, principalmente focado em desenvolvimento front-end (uma sub-área do desenvolvimento web), atividades físicas e nutrição, no passado não muito distante eu também estudava muitos assuntos ligados a economia, inclusive li alguns livros, recomendo muito o livro Crash - Uma breve história da economia.

Neste momento me sinto bastante realizado, porém ainda quero muitas coisas, mas meus objetivos são alterados de tempos em tempos, contudo no momento quero equilibrar mais minha vida, completar meus conhecimentos base em programação web e, principalmente, elaborar um projeto para ajudar pessoas a transformarem suas vidas por meio da programação de computadores (mais especificamente Desenvolvimento de sites e sistemas web - Front-end), pois minha vida, principalmente financeira, mudou por conta da computação, quero ajudar a mudar a de mais pessoas.

E este e-book é uma forma de eu conseguir ajudar pessoas a transformarem suas vidas com o uso da programação de computadores.

Além disso, minha vida fez com que eu tivesse como pontos positivos:

- a boa capacidade de aprender;
- a ideia de que consigo aprender qualquer coisa, mesmo que demore e tenha dificuldades; e
- um pequeno pluralismo de conhecimentos (principalmente informática, exercícios físicos, nutrição e economia).

Esses pontos positivos somados a minha experiência me permitem planejar um projeto para ajudar pessoas desde o início da programação até tentar vagas de emprego e isso é uma das coisas que me motivam a viver, mas não só isso, ainda sou motivado pela minha rotina, coisas que desejo fazer e, até, por todas coisas que ainda vou aprender nesta vida.

Assim, um projeto de ensino deste tipo seria uma causa que eu abraçaria, pois é uma causa social que mudou minha vida. Se um dia precisei de bicos (como segurança/garçom de festa que paga muito mal), tivemos luz "cortada" etc, hoje não tenho esses problemas.

E o melhor é que gosto muito de estudar e praticar programação de computadores, é uma forma de hobbie, pois assisto pouco a séries/filmes.

Inclusive, já participei de alguns hackathons (competições de programação), ganhei, junto a um amigo, um no Google Developer Group BH 2015 e participei do Hackathon da Rede Globo 2016.

Atualmente:

- Sou professor de disciplinas de programação (Lógica de Programação, C, C++, Web e Mobile), banco de dados e planilhas eletrônicas.
- Pesquisa sobre tecnologia no ensino, com foco em TI.
- Crio aplicações com React.js / HTML / CSS utilizando SASS (SCSS) com layouts em Grid Layout e Flexbox utilizando convenção BEM (Block-Element-Modifier) ou CSS Module.
- Aplico boas técnicas de Git Flow, gosto muito do Conventional Commits.
- Utilizo testes unitários com Jest e React Testing Library.
- Tenho conhecimento no designs de telas com Figma.
- Codifico um pouco em Typescript e Node.js com MongoDB.
- Criei e mantenho cursos de desenvolvimento web básico (HTML e CSS), Grid Layout, Programação Mobile básica (Kodular) e Excel iniciante ao intermediário no Youtube. Acesse meu canal <https://www.youtube.com/channel/UCiac8bfP7-EIP97X18IkDrQ>.

<https://www.youtube.com/channel/UCiac8bfP7-EIP97X18IkDrQ>

Quer saber mais? Acesse <http://arthurassuncao.com>

Quer me adicionar no LinkedIn? <https://www.linkedin.com/in/arthurassuncao/>

Quer ver códigos de alguns de meus projetos no Github?

<https://github.com/ArthurAssuncao>

Este sou eu e quem é você?

Capítulo 1: Por que Aprender a Programar - Ebook Programação

Capítulo 2 - Ebook Conceitos básicos

Capítulo 3: Introdução a Algoritmos

Links de dados