

UNIVERSIDADE ESTADUAL DE MARINGÁ
CENTRO DE CIÊNCIAS EXATAS
DEPARTAMENTO DE FÍSICA

SAULO HENRIQUE DOS SANTOS

**ABORDAGEM COMPUTACIONAL NO ENSINO DE FÍSICA: USO DA
LINGUAGEM PYTHON NO ENSINO MÉDIO**

MARINGÁ
2024

SAULO HENRIQUE DOS SANTOS

**ABORDAGEM COMPUTACIONAL NO ENSINO DE FÍSICA: USO DA
LINGUAGEM PYTHON NO ENSINO MÉDIO**

Monografia apresentada como parte dos requisitos necessários para aprovação do componente curricular Trabalho de Conclusão do Curso de Física.

Orientação: Prof. Dr. Anuar Jose Mincache

Coorientador: Prof. Dr. Luciano Gonsalves
Costa

MARINGÁ

2024

AGRADECIMENTOS

Agradeço, primeiramente, a Deus, por me conceder saúde, força e sabedoria ao longo desta jornada.

À minha esposa, por sua paciência, compreensão, incentivo e por cuidar de nosso filho com tanto amor, permitindo que eu me dedicasse aos estudos e à escrita deste trabalho.

Ao meu filho, cuja presença ilumina meus dias, me ensina a cada dia a valorizar pequenas conquistas e aprendizados, que renova minha motivação.

Aos meus pais e minha irmã, pelo amor incondicional, pelo apoio em todas as etapas da minha vida e por sempre acreditarem no meu potencial.

À minha sogra, pelo carinho e por cuidar do meu filho em vários momentos, garantindo que eu tivesse o tempo necessário para me concentrar neste projeto.

Aos meus professores, pela orientação, dedicação e pelo conhecimento compartilhado ao longo da graduação, que foram essenciais para a construção deste trabalho. Em especial, agradeço ao professor Dr. Anuar, por sua valiosa orientação neste projeto, pelas palavras de incentivo e pelas correções que me ajudaram a alcançar um melhor resultado.

Aos membros da banca, pelo tempo disponibilizado à leitura deste trabalho.

RESUMO

Este trabalho buscou investigar o potencial do uso da linguagem *Python* no ensino de Física no Ensino Médio, destacando sua aplicação na resolução de problemas e no desenvolvimento do pensamento computacional. Por meio da plataforma *Google Colab*, os alunos têm a oportunidade de programar algoritmos para representar fenômenos físicos, como exemplo, foram utilizadas questões de vestibulares e ENEM. A escolha do *Python* foi motivada por sua sintaxe intuitiva, versatilidade e baixa curva de aprendizado, características que tornam a linguagem ideal para iniciantes. Além disso, o *Google Colab* foi selecionado como ambiente de programação por ser uma ferramenta gratuita, acessível diretamente pelo navegador, e que oferece integração com recursos como bibliotecas pré-instaladas e colaboração em tempo real. A abordagem deste estudo visou compreender o pensamento computacional e os seus quatro pilares, a resolução de problemas na aprendizagem tradicional e na programação como um método mais dinâmico e interativo, promovendo o desenvolvimento de habilidades como decomposição, abstração, reconhecimento de padrões e algoritmos. A programação foi apresentada como uma ferramenta pedagógica que engaja os estudantes, estimulando a reflexão, o raciocínio lógico-matemático e a resolução criativa de problemas. Além de explorar questões analíticas resolvidas tradicionalmente, foram introduzidas soluções computacionais, permitindo que os alunos construam e visualizem de forma prática a aplicação dos conceitos físicos. Esse processo contribui para um aprendizado mais significativo, reforçando a importância de integrar tecnologias educacionais ao ensino de Física, apontando a programação como um recurso valioso para potencializar o engajamento e promover o desenvolvimento de competências científicas alinhadas às demandas contemporâneas.

PALAVRAS-CHAVE: Ensino de Física; Programação; Pensamento Computacional; Python; Google Colaboratory;

LISTA DE QUADROS

Quadro 1: Sugestões de inserção do PC nas disciplinas de Matemática e Ciências	11
Quadro 2: Pilares do PC aplicado ao lançamento oblíquo de projéteis	12
Quadro 3: Comparativo de sintaxe entre três linguagens de programação.....	15

SUMÁRIO

1. INTRODUÇÃO.....	7
2. FUNDAMENTAÇÃO TEÓRICA.....	8
2.1. PENSAMENTO COMPUTACIONAL	10
3. O USO DE FERRAMENTAS TECNOLÓGICAS EDUCACIONAIS.....	13
3.1. LINGUAGEM PYTHON.....	15
3.2. GOOGLE COLABORATORY (GOOGLE COLAB).....	16
4. RESOLUÇÃO DE PROBLEMAS.....	17
4.1. RESOLUÇÃO DE EXERCÍCIOS DE VESTIBULAR/ENEM NA FORMA ANALÍTICA.....	18
4.2. RESOLUÇÃO DE EXERCÍCIOS DE VESTIBULAR/ENEM USANDO PYTHON	21
CONSIDERAÇÕES FINAIS.....	26
REFERÊNCIAS BIBLIOGRÁFICAS	27

1. INTRODUÇÃO

A Física é frequentemente considerada uma disciplina desafiadora e abstrata por muitos estudantes. Essa visão se deve, em grande parte, pela complexidade de seus conceitos e pela necessidade de interpretação fenomenológica e habilidade matemática.

Uma característica da Física que a torna particularmente difícil para os alunos é o fato de lidar com conceitos abstratos e, em larga medida, contraintuitivos. A capacidade de abstração dos estudantes, em especial os mais novos, é reduzida. Em consequência, muitos deles não conseguem apreender a ligação da Física com a vida real. (Fiolhais e Trindade 2003, p.260)

No entanto, o avanço tecnológico proporciona uma oportunidade ímpar para superar tais barreiras. Fiolhais e Trindade (2003) destacam ainda a necessidade de diversificar métodos de ensino para contrariar o insucesso escolar com ajuda do computador no ensino da Física.

No cenário atual, existem dezenas de softwares, aplicativos e simuladores que podem contribuir para um processo mais efetivo do ensino-aprendizagem.

Dentre as diversas ferramentas com possibilidades educacionais, a principal ferramenta utilizada nesse trabalho é a programação. Codificar auxilia o raciocínio lógico-matemático dos estudantes, pois como afirma Silveira (2016), a programação nada mais é do que um processo de escrita, ou então a sabedoria de fazer com que o computador realize uma tarefa exatamente como queremos.

A linguagem de programação escolhida foi o *Python*, ela destaca-se por sua sintaxe intuitiva e poderosa, com uma pequena curva de aprendizado, comparativamente com outras linguagens como Java e C#, além disso, conta com uma ampla biblioteca que agiliza a programação de soluções. Não usaremos todos os recursos que a ferramenta oferece, focaremos na programação prática para a resolução de exercícios de vestibular/ENEM proposto.

A plataforma adotada para a prática da programação foi o *Google Colaboratory* (ou *Google Colab*) pela facilidade de uso diretamente no navegador, sem a necessidade de instalação. O *Colab* permite que o usuário escreva e execute códigos na linguagem *Python* e utilize suas diversas

bibliotecas pré-instaladas. É importante ressaltar que a plataforma *Google Colaboratory* possui importante integração com outros recursos oferecidos pelo Google, além do acesso facilitado e compartilhamento dos notebooks.

O presente trabalho tem o objetivo de contribuir para difundir a programação na linguagem Python, como uma ferramenta que promove uma aprendizagem ativa no ensino de Física em que o aluno pode experimentar e explorar a codificação para reforçar sua compreensão do conhecimento físico, implementando algoritmos na resolução de problemas, para isso, este trabalho está organizado em uma fundamentação teórica do pensamento computacional (PC), explorando seus conceitos e pilares, com um exemplo da aplicação do PC na física do movimento de projéteis.

Na seção sobre o uso de Ferramentas Tecnológicas Educacionais, discutimos a Linguagem *Python* e a plataforma *Google Colaboratory* (*Google Colab*), destacando suas vantagens e aplicações práticas no ambiente educacional.

Por fim, em Resolução de Problemas, apresentamos metodologias para a Resolução de Exercícios de Vestibular/ENEM na Forma Analítica e demonstramos como esses exercícios podem ser resolvidos utilizando Python, evidenciando a eficácia da programação como ferramenta de ensino e aprendizagem.

2. FUNDAMENTAÇÃO TEÓRICA

Apesar do avanço significativo nas pesquisas educacionais e das discussões sobre a necessidade de inovações no processo de ensino e aprendizagem, a prática pedagógica tradicional ainda prevalece em muitas salas de aula. Mesmo quando novas metodologias são implementadas, é comum que essas mudanças sejam temporárias, ocorrendo principalmente durante aplicações de projetos temporários de dissertações e teses ou intervenções específicas. Passado esse período, a rotina escolar tende a retornar ao modelo tradicional, caracterizado por uma abordagem expositiva centrada no professor.

O impacto da pesquisa básica em ensino de Física na sala de aulas de Física é muito pequeno, praticamente nenhum. Os resultados dessa pesquisa são publicados em revistas destinadas a pesquisadores, aos pares. Professores não participam dessa pesquisa e não leem os artigos publicados

nessas revistas. Algumas pesquisas são feitas na escola, na sala de aulas, mas apenas para gerar dados para uma dissertação, para uma tese, para um projeto específico. Uma vez feita a coleta de dados, nada muda no ensino de Física no contexto dessa coleta. (Moreira, 2018, p.74)

Outro ponto destacado por Moreira (2018) é o enfoque no ensino voltado para a testagem. Nesse contexto, os professores não se concentram no ensino aprofundado da Física, mas sim na preparação dos alunos para obterem bons resultados em exames como os vestibulares e o ENEM. Essa abordagem reflete uma visão mercadológica do ensino, evidenciada pela existência de cursinhos específicos nas escolas, direcionados exclusivamente para a preparação desses exames. O ensino para testagem não é ensino, é só treinamento para respostas de curto prazo.

É óbvio que, no contexto atual, as Tecnologias de Informação e Comunicação (TIC) deveriam permear o ensino de Física. Mas isso não acontece. Como já foi destacado, esse ensino é o mesmo de sempre: aulas expositivas e listas de problemas, quadro-de-giz (*slides em PowerPoint é a mesma coisa*), livro de texto único (ou apostila única), conteúdos desatualizados, aprendizagem mecânica (“decoreba”) de fórmulas e respostas corretas. (Moreira, 2018, p.76)

O uso de tecnologias deveria ser mais frequente e explorado no ensino, usar PowerPoint [de maneira estática] do mesmo modo que se usavam as transparências de retroprojeto no passado, não é fazer uso adequado da tecnologia.

A dinâmica tradicional reflete tanto desafios estruturais quanto culturais no sistema educacional, evidenciando a dificuldade de consolidar práticas inovadoras de forma permanente. Analisar os fatores que limitam essa transição e buscar estratégias para a continuidade das mudanças é essencial para promover um ensino mais dinâmico, significativo e alinhado às demandas contemporâneas.

É preciso pensar em como ensinar esses conteúdos, é preciso dar atenção à didática específica, à transferência didática, a como abordar a Física de modo a despertar o interesse, a intencionalidade, a predisposição dos alunos, sem os quais a aprendizagem não será significativa. (Moreira 2018, p.76)

Em outro artigo, Moreira destaca:

Aprender Física não é decorar fórmulas para resolver problemas ou definições e leis para dar respostas corretas nas provas. [...] Ensinar e aprender Física envolve conceitos e conceitualização, modelos e modelagem, atividades experimentais, competências científicas, situações que façam sentido, aprendizagem significativa, dialogicidade e criticidade, interesse. (Moreira 2021, p.2)

Frente ao cenário descrito, torna-se evidente a necessidade de uma abordagem inovadora e significativa no ensino de Física, que vá além das práticas tradicionais e do enfoque na testagem. O desafio é transformar a sala de aula em um espaço dinâmico, onde o aluno seja um agente ativo na construção do conhecimento. Neste contexto, o Pensamento Computacional e a programação emergem como ferramentas poderosas para promover uma aprendizagem mais profunda e envolvente. Integrando a resolução de problemas, a abstração e a criação de algoritmos, estas metodologias têm o potencial de transformar a experiência educacional, capacitando os alunos a desenvolverem habilidades essenciais para o século XXI. O uso do Pensamento Computacional oferece uma abordagem prática e interativa, permitindo que os estudantes compreendam e apliquem conceitos físicos de maneira inovadora e contextualizada.

2.1. PENSAMENTO COMPUTACIONAL

O pensamento computacional (PC) é o termo para a utilização de conceitos computacionais estruturados na resolução de problemas (Bordini *et al.*, 2016). O termo foi definido por Jeannette M. Wing em 2006, que estabelece o PC como modo de usar da abstração e da decomposição ao abordar uma tarefa complexa. É modelar os aspectos relevantes e escolher uma representação apropriada para um problema (Wing 2006).

Segundo (Valente 2018), apesar da popularização do termo PC, não há um consenso entre os pesquisadores sobre sua definição, as organizações *International Society for Technology in Education* (ISTE) e a *American Computer Science Teachers Association* (CSTA), identificaram nove conceitos associados ao PC: I) coleta de dados; II) análise de dados; III) representação de dados; IV) decomposição de problema; V) abstração; VI) algoritmos; VII) automação; VIII) paralelização; IX) simulação. Essas habilidades não se limitam ao campo da

ciência da computação, mas podem ser praticadas em todas as outras áreas do conhecimento. As possibilidades são inúmeras e diversificadas. Brackmann (2017) detalha, por meio de um quadro, as sugestões de inserção do PC no ensino de ciências e matemática:

Quadro 1: Sugestões de inserção do PC nas disciplinas de Matemática e Ciências

Conceitos de PC	MATEMÁTICA	CIÊNCIAS
Coleção de Dados	Encontrar uma fonte de dados de uma experiência, por exemplo: cara ou coroa, lançar dados	Coletar dados de um experimento
Análise de Dados	Contar a ocorrência de jogadas, lançamento de dados e análise de resultados	Analisar dados de um experimento
Representação de Dados	Utilizar gráfico de barras e de pizza para representação de dados. Usar conjunto, listas, representações gráficas, etc. para a visualização de informações	Resumir dados de um experimento
Decomposição de Problemas	Aplicar ordem de operadores	Realizar uma classificação de espécies
Abstração	Usar variáveis na álgebra. Estudar funções de álgebra através de comparação em computadores.	Construir um modelo de uma entidade física
Algoritmos e Procedimento	Realizar divisões longas, fatorar	Criar um procedimento experimental
Automação	Utilizar ferramentas como: Geometer, Sketch Pad, Star Logo, linhas de código em Python, etc.	Usar simulação de dados
Paralelismo	Resolução de sistemas lineares. Multiplicação de matrizes	Realizar experimentos com diferentes parâmetros simultaneamente
Simulação	Desenhar uma função em um plano cartesiano e modificar os valores das variáveis	Simular os movimentos do Sistema solar

Fonte: Brackmann (2017, p.48)

Ainda, conforme descrito por Brackmann (2017), o Pensamento Computacional utiliza-se de quatro pilares:

DECOMPOSIÇÃO, cada um desses problemas menores pode ser analisado individualmente com maior profundidade, identificando problemas parecidos que já foram solucionados anteriormente (**RECONHECIMENTO DE PADRÕES**), focando apenas nos detalhes que são importantes, enquanto informações irrelevantes são ignoradas (**ABSTRAÇÃO**). Por último, passos ou regras simples podem ser criados para resolver cada um dos subproblemas encontrados (**ALGORITMOS**). (Brackmann 2017, p.33)

O alvo do PC é o processo de resolução de problemas que consiste:

formulação de problemas de uma forma que seja possível usar um computador e outras ferramentas para ajudar a resolvê-los; organização lógica e análise de dados; representação de dados através de abstrações, como modelos e simulações; soluções de automação através do pensamento algorítmico (como uma série de passos ordenados); identificação, análise e implementação de soluções possíveis com o objetivo de alcançar a combinação mais eficiente e eficaz das medidas e recursos; generalização e transferência desse processo de resolução para problemas variados. (Bordini *et al.*, 2016, p.212).

Para ilustrar a aplicação do pensamento computacional na física, consideremos um exemplo qualitativo: a resolução de problemas de lançamento oblíquo de projéteis. A abordagem abaixo será estruturada com base nos quatro pilares do pensamento computacional: **decomposição**, **reconhecimento de padrões**, **abstração** e **algoritmos**. Esses pilares não possuem uma ordem hierárquica e são interdependentes.

Quadro 2: Pilares do PC aplicado ao lançamento oblíquo de projéteis

PILAR	APLICAÇÃO
Decomposição	Quando analisa as forças que atuam no projétil, a forma da trajetória, quando determinar as componentes vertical e horizontal do movimento.
Reconhecimento de padrões	Quando relaciona as equações do Movimento Retilíneo Uniforme (MRU) que podem ser aplicadas na componente horizontal e as equações do Movimento Retilíneo Uniformemente Variado (MRUV) que podem ser aplicadas na componente vertical, também pode analisar a aceleração no movimento que atua apenas na componente vertical.
Abstração	Quando analisa o movimento desprezando a resistência do ar, quando considera o valor da aceleração da gravidade na Terra, na Lua.
Algoritmo	Quando é construído a sequência de passos na resolução do problema com base nas equações utilizadas no cálculo do alcance, da altura máxima, do tempo de movimento.

Fonte: Elaborada pelo autor

Observa-se que, no exemplo acima, não foi explorada uma abordagem pedagógica nem apresentada uma sequência didática, aspectos que serão abordados em publicações futuras. O objetivo foi apenas demonstrar a possibilidade de integrar os pilares do pensamento computacional ao conteúdo de Física do Ensino Médio.

O pensamento computacional transforma a abordagem dos problemas, oferecendo uma metodologia estruturada e eficiente. A capacidade de decompor problemas, reconhecer padrões, abstrair detalhes irrelevantes e criar algoritmos é uma habilidade poderosa que aprimora tanto a compreensão quanto o desenvolvimento de uma mentalidade analítica para solucionar problemas.

3. O USO DE FERRAMENTAS TECNOLÓGICAS EDUCACIONAIS

A tecnologia é indispensável na sociedade atual, seu uso é crescente, tanto que as empresas mais valiosas do mundo estão no ramo de tecnologias como Apple, Microsoft e Google.

No ensino, a tecnologia potencializa as alternativas de abordagem, por meio do seu uso, possibilita enxergar o mundo não apenas como espectadores ou consumidores passivos, mas como possibilidade de um agente ativo em uma visão transformadora para projetar novas ferramentas, cultivando mentes criativas, curiosas e com capacidade de resolver problemas.

Como afirma Valente (2018), a programação, também conhecida como *coding* no contexto da Ciência da Computação, refere-se à prática de criar instruções para o computador executar com o objetivo de resolver problemas ou realizar tarefas específicas.

Ao aprender uma linguagem de programação, os alunos desenvolvem a habilidade de organizar seus pensamentos de maneira lógica e estruturada, pois desenvolve a área do cérebro responsável pelo raciocínio lógico, crítico e analítico (Silveira 2016). Essas habilidades auxiliam na organização de tarefas e até na maneira de estudar. No momento que os estudantes aprendem a programar, estão também programando para aprender. Este aprendizado permite que eles aprendam muitas outras coisas e criem novas oportunidades de aprendizagem (Brackmann 2017).

Para Borges (2021) aprender a programar abre espaço para outras aprendizagens que estão associadas à resolução de problemas, abstração e comunicação de ideias.

No caso das atividades de construção de simuladores computacionais, o processo de ensino-aprendizagem da Física pode ganhar ainda mais potencialidades. Isso porque as simulações são desenvolvidas pelo/a próprio/a usuário/a, o que torna necessário um conhecimento ainda mais aprofundado do fenômeno abordado para simular o seu comportamento. (Cruz *et al.*, 2022, p.211)

A utilização da programação no ensino de Física representa um avanço significativo em relação ao uso de simuladores prontos. Enquanto os simuladores tradicionais permitem apenas a variação de parâmetros previamente definidos, a construção de algoritmos exige do aluno um nível mais profundo de engajamento, reflexão e compreensão do conceito físico, da abordagem matemática adequada e da transcrição na construção de algoritmo, ou seja, nesse processo, o estudante precisa refletir cuidadosamente sobre os aspectos físicos envolvidos, compreender os conceitos teóricos subjacentes, trabalhar o raciocínio lógico para estruturar o desenvolvimento matemático necessário para modelar o fenômeno em questão e, posteriormente, traduzir em uma linguagem de programação.

A introdução da programação no ensino de Física, por meio da linguagem *Python*, potencializa essa abordagem ativa e a transição é simplificada desde a formulação conceitual até a implementação computacional. Comparada a outras linguagens, *Python* alia uma simplicidade na sintaxe com robustez de aplicações avançadas, o que a torna ideal para introdução no ambiente educacional, permitindo que os alunos concentrem seus esforços na compreensão dos conceitos físicos e matemáticos, ao invés de se preocuparem com detalhes técnicos da linguagem, sendo ideal para iniciantes em programação.

Convém destacar que

os recursos digitais e tecnológicos não são e não devem ser tratadas como a solução para todos os impasses presentes no ensino de Física, mas como ferramentas capazes de contribuir significativamente se usadas de maneira adequada. (Cruz *et al.*, 2022, p.212)

É importante ressaltar que o foco desse trabalho não é construir simuladores, etapa que poderá ser explorada em trabalhos futuros, mas sim de introduzir o aluno ao universo da programação, utilizando problemas físicos como instrumentos pedagógicos para esse propósito.

3.1. LINGUAGEM PYTHON

Aprender a programar pode ser desafiador, uma das principais dificuldades no aprendizado da programação de computadores reside na sintaxe complexa de várias linguagens, que exige uma maior curva de aprendizado. Nesse contexto, a linguagem de programação *Python* torna-se uma opção atraente para iniciantes devido à sua simplicidade e clareza (Menezes, 2019), reduzindo as barreiras iniciais.

Quadro 3: Comparativo de sintaxe entre três linguagens de programação

Linguagem C#	<pre>using System; class Program { static void Main() { Console.WriteLine("Hello, World!"); } }</pre>
Linguagem Java	<pre>public class HelloWorld { public static void main(String[] args) { System.out.println("Hello, World!"); } }</pre>
Linguagem Python	<pre>print("Hello, World!")</pre>

Fonte: <https://www.brasilcode.com.br/hello-world-em-30-linguagens-de-programacao-diferentes/>

Python é um software livre, uma linguagem de programação de alto nível¹ criada em 1991 por Guido Van Rossum e graças à *Python Foundation* e de inúmeros colaboradores é disponibilizada gratuitamente (Menezes, 2019).

A combinação de simplicidade com recursos poderosos e uma ampla biblioteca faz da linguagem *Python* uma das linguagens de programação mais

¹ Linguagem de alto nível: mais próxima da linguagem humana, com sintaxe mais intuitiva e legível.

populares do mundo. Como descreve Menezes (2019), “*Python* vem crescendo em várias áreas da computação, como inteligência artificial, banco de dados, biotecnologia, animação 3D, aplicativos móveis e jogos”.

Python oferece uma ampla e variada biblioteca, como *Matplotlib* (biblioteca poderosa para gerar gráficos 2D e 3D), *Numpy* (biblioteca que fornece suporte em computação científica para cálculos numéricos de grande quantidade de dados), *TensorFlow* (biblioteca voltada para computação numérica e aprendizado profundo (deep learning)), *Tkinter* (biblioteca para a construção de interfaces gráficas de usuário), *PyGame* (biblioteca projetada para criar jogos em 2D), o uso dessas e outras bibliotecas *Python* possibilitam um aumento considerável na produtividade do programador ao permitir o desenvolvimento tanto projetos simples quanto os mais complexos.

Ao ponderar o tempo disponível para o desenvolvimento do trabalho com os alunos e a rápida curva de aprendizagem, a escolha da linguagem *Python* tornou-se um fator decisivo para a realização deste projeto.

3.2. GOOGLE COLABORATORY (GOOGLE COLAB)

O *Google Colaboratory* (ou *Google Colab*) é uma plataforma online, gratuita, baseada em nuvem e executada por meio do navegador, que oferece recursos computacionais na linguagem *Python* para o uso de aprendizado de máquinas, ciência de dados, inteligência artificial e para estudos em programação *Python* pelos usuários. É um ambiente que facilita a prática de programação, mesmo em computadores com menos poder computacional, pois utiliza o poder de processamento virtual da plataforma Google e conta com diversas bibliotecas pré-instaladas.

Algumas vantagens e recursos oferecidos pelo *Google Colaboratory* (GOOGLE RESEARCH, 2024):

- Gratuito, requer apenas que o usuário tenha uma conta Google.
- Não necessita download, pois é um projeto baseado em nuvem, portanto é necessário que tenha uma conexão com a internet.
- Os notebooks do *Colab* podem ser armazenados no *Google Drive* ou *GitHub*.
- É baseado no *Jupyter notebook*, um projeto de código aberto.
- Suporte a diversas bibliotecas *Python*.

- Permite o compartilhamento com outros usuários, particularmente útil para projetos em equipe.
- O código é executado em uma máquina virtual, não sendo necessário uma máquina com hardware mais avançados e caros.
- O notebook possui divisão em blocos de texto e de código, que facilitam os comentários e a compilação do código.

O *Colab* é uma ferramenta acessível e poderosa para os estudos, sua facilidade de uso a um custo nulo, apesar da existência de uma versão PRO, faz com que seja democratizada o acesso a recursos computacionais avançados.

4. RESOLUÇÃO DE PROBLEMAS

A resolução de problemas

é a forma que o indivíduo externaliza o processo de aprender, de converter em ações, conceitos, proposições e exemplos adquiridos [...], a busca de solução de qualquer problema envolve uma readaptação do resíduo da experiência prévia frente às demandas da nova situação problemática a ser enfrentada. Se a estrutura cognitiva já possui as subsunções adequadas para permitir a reorganização do conhecimento, a resolução do problema terá cumprido o seu papel para a aprendizagem significativa. (Costa e Moreira, 2001, p.263)

A dinâmica na resolução de problemas precisa ir além de um processo mecânico de fórmula decoradas, deve ser um processo de reflexão em busca de soluções fundamentadas.

Conforme Peduzzi (1997), a resolução de problemas em Física pode ser organizada em 12 passos estratégicos que auxiliam no entendimento e solução das situações propostas, como representação visual, listagem de dados em uma análise qualitativa e quantitativa do problema.

Um dos primeiros contatos dos estudantes com a física é o conteúdo de cinemática. Desde os anos finais do ensino fundamental ou no início do ensino médio, a cinemática se faz presente em praticamente todos os materiais didáticos, sendo uma base fundamental para diversos tópicos subsequentes de Física no contexto escolar. A compreensão dos fenômenos relacionados ao movimento é facilitada quando conectada ao cotidiano dos estudantes e pode ser potencializada por ferramentas tecnológicas como simuladores, softwares gráficos e a programação. A modelagem matemática se faz presente em muitos

conteúdos de física, entender conceitos de velocidade, aceleração e outros conceitos perpassam pela construção matemática do problema. Devido a quantidade de possibilidades que podem ser exploradas, foi escolhido o tema cinemática/dinâmica para exemplificar a construção da abordagem de resolução de forma analítica e da programação.

É importante ressaltar que, comparativamente, na educação básica, a cinemática é matematicamente simplificada quando comparada com o nível superior. No presente trabalho, iremos focar no Movimento Retilíneo Uniforme (MRU) e Movimento Retilíneo Uniformemente Variado (MRUV) a nível de ensino médio, ou seja, sem aprofundamento com cálculo diferencial e integral.

Para ilustrar a aplicação da programação em *Python*, apresentarei a resolução de dois exercícios utilizando duas abordagens: A primeira abordagem consiste na resolução tradicional, empregando as equações da cinemática e da dinâmica para realizar os cálculos matemáticos de forma analítica. A segunda abordagem será realizada na linguagem de programação *Python*, com uma versão de programação mais simplificada voltada para iniciantes, demonstrando como um estudante poderia resolver o problema de forma prática e acessível, pode-se trabalhar com a interface de usuário, em que o aluno poderá programar para que dados (*input*) sejam adicionados por um usuário, mas não é o foco desse trabalho.

A construção de simulações foi planejada para ser desenvolvida em estudos futuros.

4.1. RESOLUÇÃO DE EXERCÍCIOS DE VESTIBULAR/ENEM NA FORMA ANALÍTICA

Exercício 1

(UEPG – 2023) Um móvel de massa 10 kg parte do repouso e se locomove ao longo de uma trajetória retilínea no sentido positivo do sistema de coordenadas. Sabendo que a aceleração do móvel é de 4 m/s^2 , que ele partiu de uma posição de -50 m em relação à origem e desprezando efeitos dissipativos, assinale o que for correto.

- 01) No intervalo de tempo entre $t = 0 \text{ s}$ e $t = 5 \text{ s}$, o movimento do móvel é retrógrado.
- 02) O móvel estará na origem no tempo $t = 5 \text{ s}$.

04) A força resultante aplicada no móvel é de 40 N.

08) A energia cinética do móvel quando ele se encontra na origem é de 2 kJ.

16) A distância percorrida pelo móvel no intervalo de tempo entre $t = 4$ s e $t = 6$ s é de 4 m.

Resolução

Dados

$$m = 10 \text{ kg}$$

$$v_0 = 0$$

$$a = 4 \text{ m/s}^2$$

$$S_0 = -50 \text{ m}$$

Alternativa 01 - Falsa

Para considerar um movimento retrógrado (quando se move contra a trajetória), a velocidade deve ser negativa.

Equação horária da velocidade é: $v(t) = v_0 + a \cdot t$

Substituindo os valores: $v = 0 + 4.5 \rightarrow v = 20 \text{ m/s}$

Alternativa 02 - Verdadeira

Equação horária do espaço: $S(t) = S_0 + v_0 \cdot t + \frac{a \cdot t^2}{2}$

Substituindo os valores: $S = -50 + 0.5 + \frac{4.5^2}{2} \rightarrow S = -50 + 50 \rightarrow S = 0$

Alternativa 04 - Verdadeira

Equação da 2ª Lei de Newton: $F_R = m \cdot a$

Substituindo os valores: $F_R = 10.4 \rightarrow F_R = 40 \text{ N}$

Alternativa 08 - Verdadeira

O móvel se encontra na origem no instante 5s, nesse instante a velocidade é igual a 20 m/s (calculado no item 01)

Equação da Energia Cinética: $E_C = \frac{m \cdot v^2}{2}$

Substituindo os valores: $E_C = \frac{10 \cdot 20^2}{2} \rightarrow E_C = 2000 \text{ J} \rightarrow E_C = 2 \text{ KJ}$

Alternativa 16 - Falsa

Equação horária do espaço: $S = S_0 + v_0 \cdot t + \frac{a \cdot t^2}{2} \rightarrow S = -50 + 0 \cdot t + \frac{4 \cdot t^2}{2}$

Substituindo os valores, no instante 4s: $S = -50 + 0.4 + \frac{4.4^2}{2} \rightarrow S = -18 \text{ m}$

Substituindo os valores, no instante 6s: $S = -50 + 0.6 + \frac{4.6^2}{2} \rightarrow S = 22 \text{ m}$

$$\Delta S = S - S_0 \rightarrow \Delta S = 22 - (-18) \rightarrow \Delta S = 40 \text{ m}$$

Exercício 2

(ENEM – 2023) Uma concessionária é responsável por um trecho de 480 quilômetros de uma rodovia. Nesse trecho, foram construídas 10 praças de pedágio, onde funcionários recebem os pagamentos nas cabines de cobrança. Também existe o serviço automático, em que os veículos providos de um dispositivo passam por uma cancela, que se abre automaticamente, evitando filas e diminuindo o tempo de viagem. Segundo a concessionária, o tempo médio para efetuar a passagem em uma cabine é de 3 minutos, e as velocidades máximas permitidas na rodovia são 100 km/h, para veículos leves, e 80 km/h, para veículos de grande porte. Considere um carro e um caminhão viajando, ambos com velocidades constantes e iguais às máximas permitidas, e que somente o caminhão tenha o serviço automático de cobrança. Comparado ao caminhão, quantos minutos a menos o carro leva para percorrer toda a rodovia?

- a) 30 b) 42 c) 72 d) 288 e) 360

Resolução

Dados

$$\Delta S = 480 \text{ km}$$

$$v_{\text{carro}} = 100 \text{ km/h}$$

$$v_{\text{caminhão}} = 80 \text{ km/h}$$

$$\text{Equação da velocidade média: } v = \frac{\Delta S}{\Delta t} \rightarrow \Delta t = \frac{\Delta S}{v}$$

Para o carro:

$$\Delta t_{\text{carro}} = \frac{\Delta S}{v_{\text{carro}}} \rightarrow \Delta t_{\text{carro}} = \frac{480}{100} \rightarrow \Delta t_{\text{carro}} = 4,8 \text{ h}$$

Além disso, o carro passa por 10 pedágios e fica 3 minutos em cada pedágio

$$\Delta t_{\text{pedágio}} = 10 \cdot 3 \rightarrow \Delta t_{\text{pedágio}} = 30 \text{ minutos} = 0,5 \text{ h}$$

$$\text{Totalizando } 4,8 \text{ h} + 0,5 \text{ h} = 5,3 \text{ h}$$

Para o caminhão:

$$\Delta t_{\text{caminhão}} = \frac{\Delta S}{v_{\text{caminhão}}} \rightarrow \Delta t_{\text{caminhão}} = \frac{480}{80} \rightarrow \Delta t_{\text{caminhão}} = 6h$$

A diferença de tempo será: $6h - 5,3h = 0,7h$

$$0,7h * 60 \text{ minutos} = 42 \text{ minutos}$$

Resposta: Alternativa B

4.2. RESOLUÇÃO DE EXERCÍCIOS DE VESTIBULAR/ENEM USANDO PYTHON

Exercício 1

(UEPG – 2023) Um móvel de massa 10 kg parte do repouso e se locomove ao longo de uma trajetória retilínea no sentido positivo do sistema de coordenadas. Sabendo que a aceleração do móvel é de 4 m/s^2 , que ele partiu de uma posição de -50 m em relação à origem e desprezando efeitos dissipativos, assinale o que for correto.

- 01) No intervalo de tempo entre $t = 0 \text{ s}$ e $t = 5 \text{ s}$, o movimento do móvel é retrógrado.
- 02) O móvel estará na origem no tempo $t = 5 \text{ s}$.
- 04) A força resultante aplicada no móvel é de 40 N.
- 08) A energia cinética do móvel quando ele se encontra na origem é de 2 kJ.
- 16) A distância percorrida pelo móvel no intervalo de tempo entre $t = 4 \text{ s}$ e $t = 6 \text{ s}$ é de 4 m.

```
print("Alternativa 01")
# Dados
v0 = 0 # velocidade inicial (m/s)
a = 4 # aceleração (m/s²)
t = 5 # tempo (s)

# Cálculo da velocidade
v = v0 + a * t # Velocidade final

# Verifica se o movimento é retrógrado
if v > 0:
    print("O movimento é progressivo.")
else:
    print("O movimento é retrógrado.")
```

Fonte: Elaborado pelo autor

```

print("Alternativa 02")
# Dados
S0 = -50 # posição inicial (m)
v0 = 0   # velocidade inicial (m/s)
a = 4    # aceleração (m/s²)
t = 5    # tempo (s)

# Cálculo da posição
S = S0 + v0 * t + 0.5 * a * t**2

# Verifica se o móvel está na origem
if S == 0:
    print("O móvel estará na origem no tempo t = 5 s.")
else:
    print("O móvel não estará na origem no tempo t = 5 s.")

```

Fonte: Elaborado pelo autor

```

print("Alternativa 04")
# Dados
m = 10 # massa (kg)
a = 4  # aceleração (m/s²)

# Cálculo da força
F = m * a

# Exibe o resultado
print(f"A força resultante aplicada no móvel é {F} N.")

```

Fonte: Elaborado pelo autor

```

print("Alternativa 08")
# Dados
m = 10 # massa (kg)
a = 4  # aceleração (m/s²)
t = 5  # tempo (s)

# Cálculo da velocidade
v = v0 + a * t

# Cálculo da energia cinética
E_c = 0.5 * m * v**2

# Exibe o resultado em kJ
print(f"A energia cinética do móvel na origem é {E_c / 1000:.2f} kJ.")

```

Fonte: Elaborado pelo autor

```

print("Alternativa 16")
# Dados
S0 = -50 # posição inicial (m)
a = 4    # aceleração (m/s²)

```

```
# Cálculo da posição em t = 4 s e t = 6 s
S_4 = S0 + 0.5 * a * 4**2
S_6 = S0 + 0.5 * a * 6**2

# Cálculo da distância percorrida
distancia = abs(S_6 - S_4)

# Exibe o resultado
print(f"A distância percorrida entre t = 4 s e t = 6 s é
{distancia} m.")
Fonte: Elaborado pelo autor
```

Como bônus, é possível plotar um gráfico utilizando as bibliotecas Numpy e Matplotlib

```
import numpy as np
import matplotlib.pyplot as plt

# Dados fornecidos
s0 = -50 # Posição inicial (m)
v0 = 0   # Velocidade inicial (m/s)
a = 4    # Aceleração (m/s²)
tempo_total = 6 # Tempo total (s)

# Vetor de tempo (0 a 6 segundos, com pequenos intervalos)
t = np.linspace(0, tempo_total, 500)

# Equações do movimento
velocidade = v0 + a * t # Velocidade em função do tempo
posicao = s0 + v0 * t + 0.5 * a * t**2 # Posição em função
do tempo

# Criando os gráficos
plt.figure(figsize=(12, 6))

# Gráfico da velocidade
plt.subplot(1, 2, 1)
plt.plot(t, velocidade, color="blue", label="Velocidade")
plt.title("Velocidade x Tempo", fontsize=14)
plt.xlabel("Tempo (s)", fontsize=12)
plt.ylabel("Velocidade (m/s)", fontsize=12)
plt.grid(True)
plt.legend()

# Gráfico da posição
plt.subplot(1, 2, 2)
plt.plot(t, posicao, color="green", label="Posição")
plt.title("Posição x Tempo", fontsize=14)
plt.xlabel("Tempo (s)", fontsize=12)
plt.ylabel("Posição (m)", fontsize=12)
plt.grid(True)
```

```
plt.legend()

# Mostrar gráficos
plt.tight_layout()
plt.show()
```

Fonte: Elaborado pelo autor

Exercício 2

(ENEM – 2023) Uma concessionária é responsável por um trecho de 480 quilômetros de uma rodovia. Nesse trecho, foram construídas 10 praças de pedágio, onde funcionários recebem os pagamentos nas cabines de cobrança. Também existe o serviço automático, em que os veículos providos de um dispositivo passam por uma cancela, que se abre automaticamente, evitando filas e diminuindo o tempo de viagem. Segundo a concessionária, o tempo médio para efetuar a passagem em uma cabine é de 3 minutos, e as velocidades máximas permitidas na rodovia são 100 km/h, para veículos leves, e 80 km/h, para veículos de grande porte. Considere um carro e um caminhão viajando, ambos com velocidades constantes e iguais às máximas permitidas, e que somente o caminhão tenha o serviço automático de cobrança. Comparado ao caminhão, quantos minutos a menos o carro leva para percorrer toda a rodovia?

- a) 30 b) 42 c) 72 d) 288 e) 360

```
# Dados fornecidos
distancia = 480 # km
vel_carro = 100 # km/h
vel_caminhao = 80 # km/h
tempo_cabine = 3 # minutos
num_pedagios = 10 # número de praças
# Cálculo do tempo total do carro
# Tempo de viagem na rodovia (movimento)
tempo_carro_mov = distancia / vel_carro # em horas

# Tempo gasto nos pedágios
tempo_carro_pedagio = num_pedagios * tempo_cabine / 60 # em horas

# Tempo total do carro
tempo_total_carro = tempo_carro_mov + tempo_carro_pedagio

# Cálculo do tempo total do caminhão
# Tempo de viagem na rodovia (movimento)
tempo_caminhao = distancia / vel_caminhao # em horas
```

```
# Diferença de tempo entre o carro e o caminhão
diferenca_tempo = (tempo_caminhao * 60) -
(tempo_total_carro *60)
#multiplicado por 60 para transformar em minutos

# Exibe o resultado
print(f"A diferença de tempo é de {diferenca_tempo:.2f}
minutos.")
```

Fonte: Elaborado pelo autor

CONSIDERAÇÕES FINAIS

O presente trabalho mostrou que a introdução de ferramentas tecnológicas como a linguagem Python pode transformar a forma como a Física é ensinada, tornando-a mais acessível e envolvente. O uso do Google Colab potencializa essa abordagem, eliminando barreiras técnicas e facilitando a interação entre alunos e professores. A programação aplicada à resolução de problemas físicos não só aprimora o entendimento conceitual, mas também desenvolve competências amplamente valorizadas no século XXI, como o pensamento crítico, o raciocínio lógico e a capacidade de resolver problemas.

Contudo, é importante reconhecer que a integração dessas ferramentas no ambiente escolar demanda a capacitação dos docentes e a reestruturação curricular, assegurando que a tecnologia seja usada como um complemento ao ensino, e não como um substituto. Apesar das limitações e desafios, a programação pode ser um recurso poderoso para tornar o ensino de Física mais significativo, especialmente em um contexto em que a aprendizagem tradicional apresenta diversas lacunas.

Por fim, espera-se que este trabalho inspire futuras iniciativas que explorem o uso de tecnologias na educação, especialmente na criação de simuladores e outras ferramentas interativas, ampliando o impacto dessa abordagem em diferentes áreas do conhecimento.

REFERÊNCIAS BIBLIOGRÁFICAS

BORDINI, A.; AVILA, C. M. O.; WEISSHAHN, Y.; CUNHA, M. M. da; CAVALHEIRO, S. A. da C.; FOSS, L.; AGUIAR, M. S.; REISER, R. H. S. Computação na Educação Básica no Brasil: o Estado da Arte. **Revista de Informática Teórica e Aplicada**, [S. l.], v. 23, n. 2, p. 210–238, 2016. DOI: 10.22456/2175-2745.64431. Disponível em: <https://seer.ufrgs.br/index.php/rita/article/view/RITA-VOL23-NR2-210>. Acesso em: 10 jun. 2024.

BORGES, Brunno de Oliveira. A linguagem python como ferramenta no ensino básico. 2021. 69f. Dissertação (Mestrado Profissional em Matemática em Rede Nacional) - Universidade Federal do Triângulo Mineiro, Uberaba, MG, 2021. Disponível em: <http://bdtd.ufmt.edu.br/handle/123456789/1667>. Acesso em: 11 jun. 2024.

BRACKMANN, Christian Puhlmann. Desenvolvimento do Pensamento Computacional através de atividades desplugadas na Educação Básica. 226 f. 2017. Tese de Doutorado. Tese (Doutorado em Informática na Educação) – Universidade Federal do Rio Grande do Sul, Porto Alegre. 2017. Disponível em: <http://hdl.handle.net/10183/172208>. Acesso em: 20 mai. 2024.

BRASILCODE. Hello World em 30 Linguagens de Programação Diferentes. 2023. Disponível em: <https://www.brasilcode.com.br/hello-world-em-30-linguagens-de-programacao-diferentes/>. Acesso em: 18 jun. 2024.

COLPO, R. A.; FARIA, A. U. de; MACHADO, A. F.. O ensino de física no ensino médio intermediado por programação em linguagem Python. In: ENCONTRO NACIONAL DE PESQUISA EM EDUCAÇÃO EM CIÊNCIAS, Águas de Lindóia: Abrapec, 2015. Disponível em: <https://www.abrapec.com/enpec/x-enpec/anais2015/resumos/R0575-1.PDF>. Acesso em: 17 mai. 2024.

COSTA, Sayonara Salvador Cabral da; MOREIRA, Marco Antonio. A resolução de problemas como um tipo especial de aprendizagem significativa. **Caderno Brasileiro de Ensino de Física**, [S. l.], v. 18, n. 3, p. 263–276, 2001. Disponível em: <https://periodicos.ufsc.br/index.php/fisica/article/view/6663>. Acesso em: 20 jul. 2024.

CRUZ, Clebson dos Santos; QUEIROZ GALVÃO, Lucas; ROSA, Suiane; SILVA SANTANA, Wanisson. O uso do python na construção de simuladores computacionais: proposições e potencialidades para o ensino de Física. **Caderno Brasileiro de Ensino de Física**, [S. l.], v. 39, n. 1, p. 204–237, 2022. DOI: 10.5007/2175-7941.2022.e82206. Disponível em: <https://periodicos.ufsc.br/index.php/fisica/article/view/82206>. Acesso em: 19 jun. 2024.

FIOLHAIS, C.; TRINDADE, J. Física no computador: o computador como ferramenta no ensino e na aprendizagem das Ciências Física. *Revista Brasileira de Ensino de Física*, v. 25, p. 259-272, set. 2003. Disponível em: <https://doi.org/10.1590/S1806-11172003000300002>. Acesso em: 8 mai. 2024.

GOOGLE RESEARCH. *Colaboratory FAQ*. Disponível em: <https://research.google.com/colaboratory/intl/pt-BR/faq.html>. Acesso em: 24 mai. 2024.

MÁRIO DA SILVA, Drayton; VALÉRIA FERREIRA TAVARES, Carla; MARQUES DA SILVA, Adamares. O uso da tecnologia como meio auxiliar para o ensino da Física. **CIET:EnPED**, São Carlos, maio 2018. ISSN 2316-8722. Disponível em: <https://cietenped.ufscar.br/submissao/index.php/2018/article/view/562>. Acesso em: 20 jul. 2024.

MENEZES, Nilo Ney Coutinho. Introdução à programação com Python: algoritmos e lógica de programação para iniciantes. 3. ed. São Paulo: Novatec, 2019.

MOREIRA, Marco Antonio. Desafios no ensino da Física. *Revista Brasileira de Ensino de Física* (São Paulo), v. 43, p. e20200451, 2021. Disponível em: <https://doi.org/10.1590/1806-9126-RBEF-2020-0451>. Acesso em: 13 jun. 2024.

MOREIRA, Marco Antonio. Uma análise crítica do ensino de Física. **Estudos Avançados**, São Paulo, Brasil, v. 32, n. 94, p. 73–80, 2018. DOI: [10.1590/s0103-40142018.3294.0006](https://doi.org/10.1590/s0103-40142018.3294.0006). Disponível em: <https://www.revistas.usp.br/eav/article/view/152679>. Acesso em: 13 jun. 2024.

PEDUZZI, L. O. Q.. Sobre a resolução de problemas no ensino de física. *Caderno Catarinense de Ensino de Física*, Florianópolis, v. 14, n.3, p. 229-253, 1997. Disponível em: <https://periodicos.ufsc.br/index.php/fisica/article/view/6982>. Acesso em: 10 jun. 2024.

SILVA, Martony Demes da. Aplicação da Ferramenta Google Colaboratory para o Ensino da Linguagem Python. In: ESCOLA REGIONAL DE ENGENHARIA DE SOFTWARE (ERES), 4., 2020, Evento Online. **Anais [...]**. Porto Alegre: Sociedade Brasileira de Computação, 2020. p. 67-76. DOI: <https://doi.org/10.5753/eres.2020.13717>. Acesso em: 14 jun. 2024.

SILVA, Martony Demes da. Aplicação da Ferramenta Google Colaboratory no Ensino de Ciências de Dados. In: SIMPÓSIO BRASILEIRO DE SISTEMAS COLABORATIVOS (SBSC), 16., 2021, Evento Online. **Anais [...]**. Porto Alegre: Sociedade Brasileira de Computação, 2021. p. 13-22. ISSN 2326-2842. DOI: <https://doi.org/10.5753/sbsc.2021.16017>. Acesso em: 14 jun. 2024.

SILVEIRA, L. P. N.; KNIRSCH, J. F. Como a programação pode auxiliar no desenvolvimento do raciocínio lógico em crianças, adolescentes e jovens. *Uma Nova Pedagogia para a Sociedade Futura*, Restinga Seca, RS, n. 2, p. 734–737, 2016. Disponível em: <https://ciodh.emnuvens.com.br/novapedagogia/article/view/223>. Acesso em: 18 jun. 2024.

VALENTE, J. A. Integração do pensamento computacional no currículo da educação básica: diferentes estratégias usadas e questões de formação de

professores e avaliação do aluno. Revista e-Curriculum, vol. 14, núm. 3, julho-setembro, 2016, pp. 864-897 Pontifícia Universidade Católica de São Paulo São Paulo, Brasil. Disponível em: <https://revistas.pucsp.br/index.php/curriculum/article/view/29051>. Acesso em: 8 mai. 2024.

VALENTE, J. A. Pensamento Computacional, Letramento Computacional ou Competência Digital? Novos desafios da educação. Revista Educação e Cultura Contemporânea, [S. l.], v. 16, n. 43, p. 147–168, 2019. Disponível em: <https://mestradoedoutoradoestacio.periodicoscientificos.com.br/index.php/reeduc/article/view/5852>. . Acesso em: 8 mai. 2024.

WING, J. M. “Computational thinking,” Communications of the ACM, vol. 49, pp. 33–35, Mar. 2006. Disponível em: <https://dl.acm.org/doi/10.1145/1118178.1118215>. Acesso em: 11 jun. 2024.