



# Fyks: Software de simulação gráfica com programação orientada a objetos como ferramenta didática no Ensino Médio e Fundamental

.....

Emanuela Freire Caetano Davi<sup>1</sup>,  
 Marcelo da Silva Pires<sup>1</sup>,  
 Samir Cauã Tabosa do  
 Nascimento<sup>2, #</sup>,  
 Thiago Coelho de Oliveira<sup>1</sup> e  
 Winona Ohara Freitas Rêgo<sup>1</sup>

<sup>1</sup>Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, Mossoró, Rio Grande do Norte, Brasil.

<sup>2</sup>Universidade Federal do Ceará, Fortaleza, Ceará, Brasil.

## RESUMO

Em resposta à dificuldade mostrada por parte dos alunos na compreensão de mecânica clássica e suas propriedades, um programa foi necessário para auxiliar na demonstração de fenômenos físicos utilizados para exemplificação de propriedades essenciais da dinâmica e da cinemática. Assim, desenvolvemos um software de simulação gráfica em Python para melhorar a visualização de diferentes tipos de movimento e assim auxiliar o processo de aprendizado dos alunos do Ensino Médio.

**Palavras-chave:** física no Ensino Médio; software de simulação gráfica; programação com escopo didático

.....

## 1. Introdução: a necessidade de um simulador no Ensino Médio

Surgiu a necessidade de se criar um programa capaz de representar graficamente a resolução de fenômenos físicos, principalmente nas áreas de cinemática vetorial e dinâmica de forças, sanando dúvidas recorrentes geradas pela falta de visualização dos problemas físicos em questão. A cinemática vetorial mostra-se um desafio para a compreensão de muitos alunos, uma vez que deixa o espaço bidimensional utilizado em exemplos matemáticos e utiliza como referência dois planos (x, y), que explicam as características de partículas em movimento em relação ao tempo.

De modo a aprender a base científica e as propriedades do ensino de cargas e demais assuntos de física, compreender as bases de cinemática vetorial é essencial para a aplicação prática de equações e seu comportamento na natureza. A compreensão da realidade começa com idealizações, que são conquistadas pela formação de um modelo conceitual como base. Segundo Bunge (1974), modelos são expressões esquemáticas da realidade, nos quais se inserem características que podem ser tratadas por teorias [1].

O programa, chamado de “Fyks”, foi desenvolvido na linguagem de programação Python 3, sendo feito a partir de programação orientada a objetos. A escolha do Python 3 deve-se ao fato de ser uma linguagem suportada pela maioria das plataformas presentes no mercado hoje; ela tem um código dinâmico e estável, podendo ser utilizada gratui-

tamente e em praticamente qualquer arquitetura de computadores ou sistema operacional [2], além de possuir recursos de extensibilidade que permitem integrar componentes Java e NET, assim como ser possível invocar bibliotecas das linguagens C e C++.

Ademais, o projeto foi idealizado a fim de se desenvolver um programa de simulação gráfica leve, com enfoque didático e baseado nas fórmulas físicas da mecânica clássica em seu código, tornando o programa mais rápido e dinâmico, permitindo ao usuário fácil acesso e manipulação das simulações. Nesse sentido, nota-se que um ambiente onde o aprendiz possa expressar suas idéias construindo e testando protó-

**A representação gráfica de fenômenos físicos em cinemática vetorial e dinâmica de forças ajuda a sanar dúvidas recorrentes geradas pela falta de visualização dos problemas físicos em questão**

tipos vai proporcionar tanto para os professores como para os aprendizes um espaço mais interativo e envolvente que facilitará o ensino/aprendizagem de física [3].

O público-alvo deste trabalho compreende os professores e os estudantes que possuem interesse em visualizar fenômenos da natureza como lançamentos verticais e comportamento de partículas dentro de um sistema de coordenadas e cargas interagentes no vácuo, a partir de um programa que lhes ofereça tudo isso por meio dos conceitos físicos estudados. Dessa forma, os docentes, discentes e até mesmo os interessados em física poderão desfrutar de um programa que os auxiliará no entendimento dos fenômenos físicos, de maneira que não é preciso ter conhecimentos prévios na área de programação. É necessário apenas baixar o Python 3 para visualizar com facilidade as simulações gráficas e aprender mais sobre os fenô-

#Autor de correspondência. E-mail: samir12371237@gmail.com.

menos físicos.

## 2. Referencial teórico: a mecânica clássica no software

As leis, os conceitos e as teorias da mecânica clássica fundamentam o funcionamento do *software*, como a lei da gravitação universal, as três leis da dinâmica newtoniana e os conceitos de espaço e tempo absolutos. Esse conjunto de leis foi resultado da resolução de problemas físicos que marcaram a história da física e cujas soluções desafiaram grandes mentes de sua época, como as de Galileu, Huygens e Newton, para citar apenas alguns dos mais conhecidos por alunos do ensino médio [4]. No entendimento de conceitos de cinemática e dinâmica, uma ferramenta poderosa utilizada por educadores é o uso de gráficos [5], sendo que o desenvolvimento das tecnologias para a visualização de vídeos digitais tem possibilitado ao estudante coletar e analisar dados de um experimento, gravados digitalmente, de forma rápida e eficiente, proporcionando assim a oportunidade de se envolver ativamente no processo de aprendizagem [6].

No âmbito escolar, a utilização de simuladores acessíveis promove o ensino de física, dado que as simulações contribuem de diversas formas, dependendo do grau de interação entre o estudante e o *software*, dentre as quais podemos citar: aumento da concentração dos estudantes nos experimentos, *feedback* para aperfeiçoamento do professor, geração e testes de hipóteses por parte dos estudantes, apresentação de uma versão simplificada da realidade, proporcionando melhor compreensão de conceitos abstratos etc. [7]. Ademais, é possível usar o *software* como uma ferramenta de análise de vídeo digital que pode ajudar os estudantes a desenvolver as habilidades necessárias para construir e entender gráficos cinemáticos [8].

## 3. Metodologia

### 3.1. Desenvolvimento do software

O *software* foi desenvolvido com a linguagem de programação Python 3, usando as bibliotecas Pygame e Numpy (ambas compatíveis com a versão mais recente do Python 3) para o desenvolvimento da interface gráfica e cálculos baseados em matrizes, respectivamente. Além disso, foi preciso revisar e implementar os principais conceitos da física clássica no programa, por meio

de linhas de código interdependentes. Dentre esses conceitos, foi preciso inserir grandezas vetoriais e escalares, como tempo, massa, espaço, velocidade e aceleração.

### 3.2. Utilização do software em sala de aula

No dia 29 de novembro de 2019 foi realizada uma aula de aspecto revisional na turma 2.02401.1-V do Instituto Federal de Ciência e Tecnologia do Rio Grande do Norte - Campus Mossoró acerca da seguinte subárea da física: cinemática, a qual primordialmente trabalha situações que envolvem grandezas como espaço, tempo, velocidade e aceleração. Nessa aula, foi utilizado o *software* *Fyks* a fim de proporcionar uma aula interativa aos discentes, na qual fosse possível testar se o programa facilitou o entendimento dos alunos acerca dos temas debatidos, bem como para ouvi-los a respeito de possíveis melhorias no projeto (Fig. 1).

No primeiro momento, foram retomados os conceitos da cinemática e, posteriormente, foram resolvidas três questões sobre o assunto; depois disso, os fenômenos trabalhados nas questões foram apresentados por meio do programa, simulando-os graficamente, com o intuito de complementar a visua-

lização do problema abordado. Em seguida, foi entregue um questionário aos alunos para que respondessem se o programa facilitou ou não o entendimento dos problemas e quais as possíveis melhorias que poderiam ser realizadas para aperfeiçoar esse trabalho.

A utilização do programa foi essencialmente auxiliar, complementando visualmente o que foi perguntado nas três situações-problema propostas (Apêndice A); desse modo, partindo-se com uma abordagem explicativa, procurou-se explorar o uso do programa na resolução de questões. Diante disso, o docente Marcelo Pires aproveitou o simulador como um instrumento didático, possibilitando a complementação do entendimento para os discentes e proporcionando um aprendizado significativo.

## 4. Resultados

### 4.1. Principais funcionalidades do *Fyks*

Foi obtido um produto didático capaz de representar e apresentar fenômenos físicos com clareza (Fig. 2). Essa funcionalidade fica evidente quando se evidencia a flexibilidade que o programa permite, tendo em vista que é possível alterar os valores das grandezas físicas com extrema

**No âmbito escolar, a utilização de simuladores acessíveis promove o ensino de física, aumentando a concentração dos estudantes nos experimentos**



Figura 1 - Realização da aula experimental proposta pelo docente Marcelo Pires, na sala de aula 01 no IFRN - campus Mossoró (2019).



Figura 2 - Print proveniente de computador aberto no programa PyCharm com o *software* de simulação gráfica instalado e aberto (2020).

facilidade, sendo que com um clique na caixa de seleção o usuário consegue modificar o valor desejado.

Nesse sentido, é possível realizar alterações em parâmetros como velocidade inicial, altura inicial, aceleração gravitacional e escolher se haverá ou não presença da resistência do ar no sistema em determinados corpos. Dessa forma, o usuário tem liberdade de optar por usar  $9,8 \text{ m/s}^2$  ou  $10 \text{ m/s}^2$  como valor da gravidade na Terra no seu experimento virtual, assim como pode usar a resistência do ar, caso queira fazer outro tipo de simulação envolvendo cinemática ou dinâmica vetorial.

Assim, o programa consegue contemplar tanto os docentes de ensino fundamental, ao possibilitar a utilização de valores ideais (ou até arredondados, para a simplificação do problema), quanto os do Ensino Médio, ao propiciar o uso de valores mais próximos da realidade. Por conseguinte, o programa proporciona o manuseio dos valores da forma como o usuário sentir necessidade, gerando uma melhor visualização dos fenômenos físicos.

Em novembro de 2019, a maioria dos estudantes de Ensino Médio que usaram o programa deram uma avaliação positiva em relação ao uso do *soft-*

*ware* em sala de aula, sendo que todos os 28 estudantes que participaram da aula responderam “Sim” para a pergunta “A aula interativa foi capaz de complementar o conteúdo ministrado pelo professor?” (Apêndice B).

#### 4.2. Requisitos físicos para o uso do Fyks em sala de aula

A exibição do *software* em sala de aula pode ocorrer por meio de um projetor ou monitor de resolução média ou alta. Quanto aos requisitos mínimos de utilização no computador, são eles: 2GB de memória RAM e um processador Intel Pentium T4400 (2.20 GHz) com sistema operacional de 32 bits. Entretanto, o recomendado é um computador com um processador Intel Core i3 de sexta geração com placa de vídeo integrada (ou versões superiores) e uma memória RAM de 4 GB (ou superior).

#### 5. Considerações finais

Foi construído um programa de simulação gráfica que se encontra disponível de modo on-line na plataforma GitHub pelo link: <https://rabbithy.github.io/Fyks/>. Sendo assim, é um produto gratuito e de fácil acesso, que pode ser utilizado como ferramenta didática para o ensino de cinemática e dinâmica no Ensino Médio, esclarecendo possíveis dúvidas sobre os assuntos. Ademais, o código também é livre para ser consultado para possíveis dúvidas acerca do funcionamento do programa.

### Apêndices

#### Apêndice A. Questões de cinemática propostas pelo docente Marcelo Pires

1. Dois automóveis colidem em um cruzamento. O veículo A estava a  $20 \text{ m/s}$  nos  $100 \text{ m}$  antes da colisão, conforme apontado por fotossensores na via. Enquanto A passava a  $100 \text{ m}$  do cruzamento, B saía de um semáforo a  $75 \text{ m}$  deste, com aceleração constante. Com que velocidade B estava no momento da colisão?
2. Um automóvel A encontra-se em repouso diante de um semáforo fechado. Assim que o semáforo abre, A está entrando em movimento e outro automóvel B está passando por ele. O gráfico abaixo mostra as velocidades de A e B em função do tempo. Em que instante T os automóveis voltam a se encontrar?
3. Um objeto lançado do alto de um prédio com velocidade  $12 \text{ m/s}$ , horizontal, atinge o solo a que distância do prédio, considerando que a altura deste é de  $125 \text{ m}$ ?

Aceleração da gravidade:  $g = 10 \text{ m/s}^2$ .

Disponível de modo completo em: <https://bit.ly/2uLUSJO>

#### Apêndice B. Questionário destinado aos discentes sobre a avaliação da aula experimental

1. A aula interativa foi capaz de complementar o conteúdo ministrado pelo professor?
2. A aula tradicional fazendo simulações no quadro branco é mais simples de compreender ou o simulador computacional facilitou o entendimento?
3. Quais são as possíveis melhorias que o simulador poderia ter?
4. É possível visualizar facilmente os experimentos?
5. Comentários livres a respeito da aula.

Disponível de modo completo em: <https://bit.ly/3a6HRtw>

## References

- [1] M.A. Bunge, *Teoria e Realidade* (Perspectiva, São Paulo, 1974), v. 1, p. 25.
- [2] M.N.N. Coutinho, *Introdução à Programação com Python* (Novatec, São Paulo, 2014), v. 2, p. 25.
- [3] C.F. Santos, C.S. Menezes, in: *Anais do XI Workshop de Informática na Escola*, São Leopoldo, 2005, editado por J.A.C. Filho (Sociedade Brasileira de Computação, São Leopoldo, 2005), p. 2747.
- [4] D.D. Uzêda, *Tópicos em Mecânica Clássica*. Dissertação de Mestrado, Universidade Federal do Rio de Janeiro, 2011.
- [5] V.B. Barbeta, I. Yamamoto, *Revista Brasileira de Ensino de Física* **24**, 158 (2002). <http://dx.doi.org/10.1590/S1806-11172002000200012>
- [6] M. Kearney, D.F. Treagust, *Austr. J. of Educ. Techn.* **17**, 68 (2001). <https://doi.org/10.14742/ajet.1773>
- [7] A.A. Soares, L.E. Moraes, F.G. Oliveira, *Cad. Bras. Ens. Fís.* **32**, 917 (2015). <https://doi.org/10.5007/2175-7941.2015v32n3p915>
- [8] R.J. Beichner, *Am. J. Phys.* **64**, 1272-1277 (1996). <https://doi.org/10.1119/1.18390>