

# Modelagem computacional – uma breve introdução<sup>1</sup>

## Parte 1: desenhando objetos, linhas e vetores

A modelagem computacional, hoje em dia, joga um papel fundamental em praticamente todos os campos da ciência e da engenharia. Na física é um instrumento essencial de pesquisa tanto em física teórica como experimental. Criar modelos computacionais simples, baseados em princípios físicos fundamentais, permite que vejamos mais claramente como esses princípios governam o comportamento dos sistemas reais. Esses modelos permitem ainda que apliquemos esses princípios a sistemas muito mais complicados (e realistas) que aqueles poucos que podemos resolver analiticamente com o auxílio do cálculo diferencial e integral. Ferramentas computacionais adequadas permitem ainda que visualizemos a evolução temporal e o comportamento de sistemas em 3D, o que ajuda a entender o significado físico de grandezas vetoriais como velocidade e momento.

### **E se eu não sei nada de programação?**

Não se preocupe, nenhum conhecimento prévio em programação é necessário (assim como nenhum conhecimento prévio de cálculo). Você não terá que produzir programas completos e aprenderá tudo o que é necessário para construir programas simples, mas muito poderosos, ao longo da disciplina. Esses programas serão usados para entender melhor os conceitos físicos e a matemática subjacente aos conteúdos que estão sendo ensinados.

### **Por que aprender computação em um curso introdutório de física?**

#### **Não é perda de tempo?**

A mecânica clássica, teoria proposta por Newton e que descreve como e por que os corpos se movem, está ligada de forma indissolúvel a um ramo da matemática conhecido como cálculo diferencial e integral. O “cálculo” – como geralmente é chamado – foi desenvolvido (entre outros) pelo próprio Newton que precisava dessa matemática para escrever sua teoria. Tradicionalmente as primeiras semanas de um curso introdutório de física são gastas com uma introdução informal e superficial a algumas poucas técnicas de derivação e integração e a uma revisão de cinemática unidimensional, na qual essa matemática é exemplificada. A ênfase em problemas unidimensionais, com forças e acelerações constantes ou polinomiais prossegue, depois, não por serem esses problemas fundamentais na física, mas por serem os únicos que podem ser resolvidos analiticamente com essa introdução ao cálculo superficial e simplificada. Paga-se um preço alto por isso: a complexidade e o poder preditivo das ideias de Newton é, de certa forma, escamoteado dos alunos, até que o curso de cálculo avance o suficiente, o que pode levar quase todo o semestre. Passa-se um tempo grande aprendendo problemas às vezes artificiais ou sem muita relevância para a ciência de hoje, apenas porque neles sabemos fazer as contas. Derivadas e integrais complicadas, porém, podem ser resolvidas facilmente no computador, com noções de computação mais simples de ensinar e aprender que as de cálculo, permitindo que se discuta desde o primeiro dia problemas do mundo real, que é tridimensional, com forças variáveis e complicadas. Mais à frente, depois de algumas semanas, quando as técnicas analíticas do cálculo já tiverem sido apresentadas de forma adequada nessa disciplina, o tratamento analítico será retomado de modo que todo o conteúdo usual seja visto. Ainda, como nos dias de hoje resolver no computador problemas complicados é rotina em qualquer área de fronteira da física, vocês terão, desde o início, uma ideia bem mais completa do que é ser um cientista e fazer ciências na área de exatas.

### **O que é o VPython?**

Um programa de computador é escrito em uma “*linguagem de programação*”, que nada mais é do que um conjunto de palavras e regras necessárias para comunicar instruções a um computador. Essas linguagens obedecem a determinadas “*sintaxes*”, ou seja, além das palavras certas, as instruções devem obedecer certas regras (pontuação, ordenamento, etc.) para que sejam entendidas. Ainda, para escrever um programa de computador necessitamos programas de edição de textos específicos para essa função (em vez de programas como Word ou LibreOffice).

Escolhemos a linguagem VPython para construir os programas dessa disciplina. VPython é uma extensão do

---

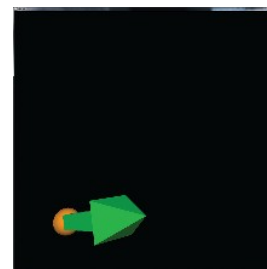
<sup>1</sup> Texto escrito a partir de algumas seções do livro *Matter and Interactions*, vol. 1, R. Chabay e B. Sherwood, Wiley.

Python, uma linguagem bastante conhecida de programação. Ele vem preparado para a álgebra vetorial em 3D e permite de forma muito simples criar animações como consequência dos cálculos realizados a partir dos princípios físicos. É uma linguagem aberta, gratuita e funciona nos sistemas operacionais Windows, Linux ou MacOS. Vocês podem facilmente instalá-lo em um computador pessoal, ou ainda executar os programas através da Internet sem instalar nada. No final desse texto damos algumas dicas para essa instalação, que é simples. Como editor de texto usaremos o *Geany*, também gratuito e de fácil instalação em qualquer desses sistemas operacionais. Há outros editores possíveis (como o VIDE, que costuma vir junto com o VPython, e é até mais simples de usar). Nos computadores disponíveis na sala de aula 212 usaremos preferencialmente o sistema operacional Linux. Também não é necessário ter familiaridade com o Linux para acompanhar as aulas.

A seguir, mostramos um exemplo de um pequeno programa em VPython. Uma visão da tela produzida por esse programa está na figura ao lado.

### Código

```
from visual import *  
bola = sphere(pos = vector(-3,-3,0), color = color.orange, radius = 0.5)  
velocidade = vector(3,1.5,4)  
arrow(pos = bola.pos, axis = velocidade, color = color.green)
```



No código acima, uma esfera denominada “bola” é criada na posição  $\langle -3; -3; 0 \rangle$ . A partir dessa informação, tente localizar, na figura acima, onde está a origem do sistema de coordenadas empregado.

A origem do sistema de coordenadas,  $\langle 0; 0; 0 \rangle$ , está no centro do quadrado; o eixo x é horizontal e aponta para a direita, o eixo y é vertical e aponta para cima e o eixo z é perpendicular à tela e aponta na sua direção. A localização da esfera laranja, designada por “bola” no programa, é consistente, nesse sistema de coordenadas, com a posição  $\langle -3; -3; 0 \rangle$ .

### O que poderíamos alterar no código para criar uma esfera maior?

A esfera chamada “bola” é criada com raio igual a 0,5. Para alterá-lo teríamos que informar um raio maior.

### Quanto de programação vou ter que aprender?

Muito pouco. Os programas virão praticamente prontos. Algumas linhas deverão ser completadas com instruções envolvendo diretamente os *conceitos* que estão sendo estudados. Como calcular o vetor posição relativa? Esta é um exemplo de uma pergunta a ser respondida. Primeiro é necessário discutir e verificar se todos na equipe entenderam o *conceito* de vetor posição relativa; depois, *que contas devemos fazer* para obtê-lo. Só então vocês irão se preocupar em como instruir o computador a fazer essas contas – e vamos dar todas as orientações necessárias a isso. Quando fornecemos ao computador instruções precisas dessas contas e executamos o programa para que ele de fato as faça, pode-se, imediatamente, de forma visual, verificar se o raciocínio e os conceitos empregados estão corretos. Não será necessário nada mais que isso. Mas pode ser motivador tentar fazer bem mais!

É muito mais fácil aprender a escrever programas de computador em VPython assistindo vídeos e vendo os resultados na tela imediatamente. Se tiver interesse em se aprofundar, há uma série de tutoriais no Youtube (*VPython instructional videos 1: 3D objects*, em [vpython.org/video01.html](http://vpython.org/video01.html) por ex.) que foram desenvolvidos para ajudar estudantes de física a aprenderem VPython. Alguns problemas do livro texto (problemas P68 – P77 na 4ª ed.) pedem que assistam aos vídeos e depois executem os desafios propostos neles. O VPython pode ser baixado e instalado a partir do site <http://vpython.org> ; siga as instruções do site. Teremos prazer em ajudá-los, se necessário. Também, é possível executar e editar um programa em VPython em um computador sem instalar nada, desde que o computador esteja conectado com a Internet. Basta ir ao site [www.glowscript.org](http://www.glowscript.org) . Uma vez no site, você deve fazer o login (sign in); crie um usuário e senha se necessário, seguindo as instruções. Após o login vai aparecer uma tela de boas vindas; na quarta linha a frase “ You are signed in as XXX and your programs are here. Entre no link “here” e boa sorte!

Esse recurso permite que todos os alunos possam revisar em casa o que não entenderam direito na aula.