

Programando sob as leis da natureza

* Toni Oliveira

The Nature of code, de Daniel Shiffman, é uma poderosa ferramenta para os interessados em simulações computacionais de sistemas naturais. Trata-se de um livro de programação, em nível não introdutório, no qual diversos fenômenos naturais são interpretados de forma algorítmica e implementados em linguagem de programação. O autor aborda desde aplicações das leis da Física Newtoniana sobre objetos inanimados, até a criação de algoritmos de simulação de redes neurais. O objetivo geral do livro, segundo o próprio Shiffman (2012, p. ix), é tornar o leitor capaz de determinar como escrever um código de computador que simule uma ocorrência natural de nosso mundo físico.

A ideia de escrever o livro surgiu da experiência de Shiffman em oferecer, durante cinco anos, como professor do ITP¹ (Interactive Telecommunications Program) da Tisch School of the Arts na New York University, um curso sobre o tema de simulação de sistemas naturais na linguagem Processing. Assim, ele resolveu utilizar o mesmo nome do curso na publicação e, conseqüentemente, a maior parte do material dessas aulas como matéria prima para a obra.

¹Website do programa: <<https://tisch.nyu.edu/itp>>

* Toni Oliveira., mestrando do Programa Multidisciplinar de Pós-Graduação em Cultura e Sociedade na linha de pesquisa Cultura e Arte da Universidade Federal da Bahia



A origem do conteúdo, a partir de uma aplicação didática prática, conferiu ao texto objetividade e simplicidade muito úteis na apresentação de um tema tão complexo. Essas são características do autor, já amplamente reconhecidas por seus alunos² da NYU, assim como pelos seguidores nos diversos canais que ele mantém na rede, para ensino de programação na linguagem Processing³ e P5.js⁴. Shiffman tornou-se, em 2012, um terceiro líder, ao lado de Casey Reas e Ben Fry, no projeto da Processing Foundation e todos os exemplos de código do livro estão nessa linguagem.

Sobre sua escolha pela linguagem de programação *Processing*, ele justifica por tratar-se de um projeto de software livre e código aberto, além de ser a linguagem que o faz sentir-se mais confortável para utilização em seus projetos pessoais. Na verdade, essas ideias têm muita correspondência com a forma de publicação escolhida por Shiffman e que comentarei ao final dessa resenha.

Apesar desse direcionamento ao *Processing*, todos os conceitos por trás dos algoritmos apresentados no livro são perfeitamente reutilizáveis para seu desenvolvimento em outras linguagens de programação, pois o autor comenta e explica detalhadamente cada tópico antes de transcrevê-los em código. As explicações dos fenômenos naturais, bem como da matemática necessária aos raciocínios implementados nos programas, demonstram claramente a intenção do autor em estimular o aprendizado consciente das concepções algorítmicas empregadas nos exercícios desenvolvidos.

Mesmo com todo o apoio didático minucioso presente no texto, ter experiência prévia de programação com orientação a objeto é indispensável para a compreensão da maioria dos códigos do livro. Shiffman sugere no prefácio que, caso o leitor ainda não possua essa proficiência em programação, dedique alguns meses de estudo ao *Processing* antes de dar seguimento à leitura do *Nature of code*. Uma boa alternativa é um outro livro escrito pelo mesmo autor, o *Learning Processing*, considerado como o ponto de partida na aprendizagem de muitos atuais programadores em *Processing*.

²Os alunos de Shiffman mantêm um website - Dan Shiffman is Awesome – onde compartilham depoimentos sobre ele e trabalhos artísticos realizados com sua ajuda ou sob sua tutoria. Disponível em: <<http://www.danshiffmanisawesome.com/>>

³<<https://processing.org/>>

⁴<<http://p5js.org/>>



Shiffman relata no prefácio, que sua experiência como professor, o levou a perceber, que os alunos permaneciam mais interessados quando desenvolviam um único projeto ao longo do semestre de aulas. Por conta disso, sua proposta é que o leitor desenvolva um projeto que acompanhe gradualmente os tópicos dos capítulos do livro. Esse projeto é chamado de *The Digital Ecosystem* e compreende um mundo de criaturas virtuais com comportamentos próprios para ser projetado em uma tela na entrada de um museu. Ao final de cada capítulo, o projeto fictício proposto é atualizado através dos exercícios de programação.

O livro contém dez capítulos divididos em três partes principais, além de um prefácio também dividido em oito partes, que explicam ao leitor o que ele precisa saber e como tirar o maior proveito do conteúdo da publicação. Shiffman ainda elaborou um capítulo de introdução no qual explica o conceito de aleatoriedade, probabilidade, apresenta o algoritmo do Ruído de Perlin e uma rápida definição de objetos e classes no contexto do paradigma da Programação Orientada a Objeto (POO).

A primeira parte, *Inanimate Objects*, compreendida pelos primeiros cinco capítulos, trata exatamente das simulações da Mecânica Newtoniana sobre objetos inanimados. Shiffman nos faz experimentar o Espaço Euclidiano através do uso dos vetores já implementados em uma classe nativa do Processing (*PVector Class*) para construir simulações de forças, efeitos de oscilação e sistemas de partículas. A classe *PVector* do Processing é muito utilizada em todo o livro, pois através dela, o acesso à matemática vetorial dispensa conhecimentos mais aprofundados.

Ao fim da primeira parte, no quinto capítulo (*Physics Libraries*), o autor apresenta duas poderosíssimas bibliotecas de desenvolvedores não ligados a *Processing Foundation: Box2D* e *Toxiclibs*. A *Box2D* é uma biblioteca desenvolvida originalmente em C++ para se trabalhar em um ambiente físico de maior correspondência com o mundo real. O sistema de coordenadas da *Box2D* não é o padrão de *pixel world* dos softwares e seu centro referencial é o meio do plano representado na tela. Ela possibilita os recursos de colisão entre formas geométricas, conexões (elásticas, de articulações, com polias e engrenagens), motores (movimento) e fricção. A *Toxiclibs* tem como principal característica a fácil implementação do Método de Integração de Verlet, que integra as equações de movimento de Newton na criação de ambientes de



partículas em simulações computacionais. Seus destaques em relação a Box2D são a criação de ambiente físico tridimensional e simulação das forças de atração e repulsão entre partículas.

Na segunda parte do livro, o autor começa a modelar um ambiente com entidades vivas. No capítulo seis, os objetos, até então inanimados, adquirem status de agentes autônomos, ou seja, entidades com capacidades de ações próprias independentes em um meio ambiente. Os algoritmos apresentados nesse capítulo tornam esses agentes capazes de perceberem o meio, processar as informações e calcular suas ações. No final desse capítulo, Shiffman ainda acrescenta dois importantes tópicos sobre eficiência algorítmica e dicas de otimização, conhecimentos valiosos que os programadores mais experientes nem sempre compartilham.

Adiante, o capítulo sete aborda a criação de modelos de sistemas de células dispostas em grade e com atribuição de estados que influenciam, tanto as mais próximas, quanto as pertencentes as gerações subsequentes. Finalizando o segundo bloco, o assunto é a geometria da natureza através da geração de padrões a partir de fractais e com utilização de algoritmos recursivos.

Na terceira parte, *Intelligence*, Shiffman explica os algoritmos genéticos e suas etapas: criação de uma população, aplicação do Princípio de Seleção de Darwin e reprodução. Em seguida, no final do capítulo nove, é feita a codificação desses algoritmos para criação de um ecossistema simulado.

O décimo capítulo finaliza a terceira parte abordando os sistemas de redes neurais na computação. Após uma conceituação bastante clara de um *perceptron* (algoritmo que decide se uma entrada pertence a uma ou outra classe estabelecida), Shiffman conduz o passo a passo detalhado para a codificação dessas redes e algumas visualizações de fluxo de dados através delas.

Ao final do livro o leitor já possui todos os conhecimentos necessários a conclusão do *The Digital Ecosystem* e a maior qualidade desse livro é justamente a forma didática aplicada ao tema por Daniel Shiffman. Os passos para construção de um projeto permitem experimentações imediatas dos conteúdos teóricos em um contexto orientado a um resultado maior e específico. Dessa forma, toda a complexidade do tema é bastante diluída e, quando chegamos ao fim do livro, nos damos conta da quantidade considerável de conceitos científicos e técnicas de



programação que experimentamos.

The Nature of code pode ser adquirido nos formatos impresso tradicional ou digital, em livrarias ou direto com o autor. Shiffman decidiu ter maior controle sobre o processo de publicação e editoração de sua obra desde que se sentiu frustrado com o lançamento do *Learning Processing* em 2008. Por conta disso, ele disponibilizou diversas formas de acesso ao livro como o sistema de compra “pague o quanto quiser”, em seu próprio *website*, que permite até mesmo o download de graça de uma versão em PDF. Há também um repositório no *Github* com todos os códigos periodicamente atualizados e revisados, tanto por Shiffman, quanto pelos leitores, além de um arquivo ASCIIDOC para geração de outras versões eletrônicas (PDF, HTML, EBooks) através da ferramenta de publicação *open source Magic Book Project* permitindo assim a atualização constante do livro.

Links:

Site do livro: <<http://natureofcode.com/>>

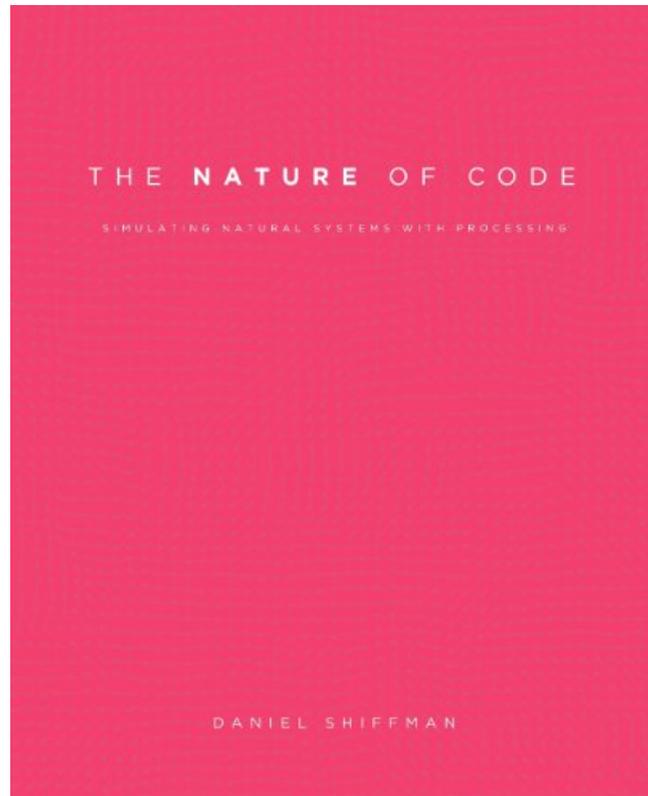
Repositório de códigos: <<https://github.com/shiffman/The-Nature-of-Code>>

Site do autor: <<http://shiffman.net/>>

Ficha técnica:

1. Título: *The Nature of Code: Simulating Natural Systems with Processing*
2. Autor: Daniel Shiffman
3. Ilustrador: Zannah Marsh
4. Capa: David Wilson
5. Nº de edição: 1ª
6. Idioma: Inglês
7. Número de páginas: 520
8. Editor: Shannon Fry
9. Local de edição: CA - EUA
10. Data de edição: 13 de dezembro de 2012
11. Tiragem (não disponível, versão online)
12. Data de impressão: 2012
13. ISBN-10: 0985930802
14. ISBN-13: 978-0985930806





The Nature of Code: Simulating Natural Systems with Processing - Daniel Shiffman

