

Creative Code Aesthetics Computation

Ten laws of simplicity for business, technology, and design that teach us how to need less but get more. Finally, we are learning that simplicity equals sanity. We're rebelling against technology that's too complicated, DVD players with too many menus, and software accompanied by 75-megabyte "read me" manuals. The iPod's clean gadgetry has made simplicity hip. But sometimes we find ourselves caught up in the simplicity paradox: we want something that's simple and easy to use, but also does all the complex things we might ever want it to do. In *The Laws of Simplicity*, John Maeda offers ten laws for balancing simplicity and complexity in business, technology, and design—guidelines for needing less and actually getting more. Maeda—a professor in MIT's Media Lab and a world-renowned graphic designer—explores the question of how we can redefine the notion of "improved" so that it doesn't always mean something more, something added on. Maeda's first law of simplicity is "Reduce." It's not necessarily beneficial to add technology features just because we can. And the features that we do have must be organized (Law 2) in a sensible hierarchy so users aren't distracted by features and functions they don't need. But simplicity is not less just for the sake of less. Skip ahead to Law 9: "Failure: Accept the fact that some things can never be made simple." Maeda's concise guide to simplicity in the digital age shows us how this idea can be a cornerstone of organizations and their products—how it can drive both business and technology. We can learn to simplify without sacrificing comfort and meaning, and we can achieve the balance described in Law 10. This law, which Maeda calls "The One," tells us: "Simplicity is about subtracting the obvious, and adding the meaningful."

The New Aesthetic and Art: Constellations of the Postdigital is an interdisciplinary analysis focusing on new digital phenomena at the intersections of theory and contemporary art. Asserting the unique character of New Aesthetic objects, Contreras-Koterbay and Mirocha trace the origins of the New Aesthetic in visual arts, design, and software, find its presence resonating in various kinds of digital imagery, and track its agency in everyday effects of the intertwined physical world and the digital realm. Contreras-Koterbay and Mirocha bring to light an original perspective that identifies an autonomous quality in common digital objects and examples of art that are increasingly an important influence for today's culture and society.

The book explores the technical as well as cultural imaginaries of programming from its insides, demonstrating the reflexive practice of aesthetic programming, to understand and question existing technological objects and paradigms.

Considering how culturally indispensable digital technology is today, it is ironic that computer-generated art was attacked when it burst onto the scene in the early 1960s. In fact, no other twentieth-century art form has elicited such a negative and hostile response. When *the Machine Made Art* examines the cultural and critical response to computer art, or what we refer to today as digital art. Tracing the heated debates between art and science, the societal anxiety over nascent computer technology, and the myths and philosophies surrounding digital computation, Taylor is able to identify the destabilizing forces that shape and eventually fragment the computer art movement.

Processing

Redesigning Leadership

Constellations of the Postdigital

The Philosophy of Creativity

Posthumanity: Merger and Embodiment

From Foundational Exercises to Theories of Representation

Creative Code

Nietzsche was not interested in the nature of art as such, or in providing an aesthetic theory of a traditional sort. For he regarded the significance of art to lie not in l'art pour l'art, but in the role that it might play in enabling us positively to 'revalue' the world and human experience. This volume brings together a number of distinguished figures in contemporary Anglo-American Nietzsche scholarship to examine his views on art and the aesthetic in the context of this wider philosophical project. All of the major themes of Nietzsche's aesthetics are discussed: art and the affirmation of life, the relationship between art and truth, music, tragedy, the nature of aesthetic experience, the role of art in Nietzsche's positive ethics, his critique of romanticism, and his ambivalent attitude towards Richard Wagner.

Lessons for a new generation of leaders on teamwork, meetings, conversations, free food, social media, apologizing, and other topics. When designer and computer scientist John Maeda was tapped to be president of the celebrated Rhode Island School of Design in 2008, he had to learn how to be a leader quickly. He had to transform himself from a tenured professor—with a love of argument for argument's sake and the freedom to experiment—into the head of a hierarchical organization. The professor is free to speak his mind against "the man." The college president is "the man." Maeda has had to teach himself, through trial and error, about leadership. In *Redesigning Leadership*, he shares his learning process. Maeda, writing as an artist and designer, a technologist, and a professor, discusses intuition and risk-taking, "transparency," and all the things that a conversation can do that an email can't. In his transition from MIT to RISD he finds that the most effective way to pull people together is not social networking but free food. Leading a team? The best way for a leader to leverage the collective power of a team is to reveal his or her own humanity. Asked if he has stopped designing, Maeda replied (via Twitter) "I'm designing how to talk about/with/for our #RISD community." Maeda's creative nature makes him a

different sort of leader—one who prizes experimentation, honest critique, and learning as you go. With *Redesigning Leadership*, he uses his experience to reveal a new model of leadership for the next generation of leaders.

Contingent Computation offers a new theoretical perspective through which we can engage philosophically with computing. The book proves that aesthetics is a viable mode of investigating contemporary computational systems.

Finally, a book on creative programming, written directly for artists and designers! Rather than following a computer science curriculum, this book is aimed at creatives who are working in the intersection of design, art, and education. In this book you'll learn to apply computation into the creative process by following a four-step process, and through this, land in the cross section of coding and art, with a focus on practical examples and relevant work structures. You'll follow a real-world use case of computation art and see how it relates back to the four key pillars, and addresses potential pitfalls and challenges in the creative process. All code examples are presented in a fully integrated Processing example library, making it easy for readers to get started. This unique and finely balanced approach between skill acquisition and the creative process and development makes *Coding Art* a functional reference book for both creative programming and the creative process for professors and students alike. What You ' ll Learn Review ideas and approaches from creative programming to different professional domains Work with computational tools like the Processing language Understand the skills needed to move from static elements to animation to interaction Use interactivity as input to bring creative concepts closer to refinement and depth Simplify and extend the design of aesthetics, rhythms, and smoothness with data structures Leverage the diversity of art code on other platforms like the web or mobile applications Understand the end-to-end process of computation art through real world use cases Study best practices, common pitfalls, and challenges of the creative process Who This Book Is For Those looking to see what computation and data can do for their creative expression; learners who want to integrate computation and data into their practices in different perspectives; and those who already know how to program, seeking creativity and inspiration in the context of computation and data.

Critical Code Studies

The Four Steps to Creative Programming with the Processing Language

When the Machine Made Art

Animation and New Media Art

Aesthetic Computing

A Little-Known Story about a Movement, a Magazine, and the Computer's Arrival in Art

Phantasmal Media

Geocomputation with R is for people who want to analyze, visualize and model geographic data with open source software. It is based on R, a statistical programming language that has powerful data processing, visualization, and geospatial capabilities. The book equips you with the knowledge and skills to tackle a wide range of issues manifested in geographic data, including those with scientific, societal, and environmental implications. This book will interest people from many backgrounds, especially Geographic Information Systems (GIS) users interested in applying their domain-specific knowledge in a powerful open source language for data science, and R users interested in extending their skills to handle spatial data. The book is divided into three parts: (I) Foundations, aimed at getting you up-to-speed with geographic data in R, (II) extensions, which covers advanced techniques, and (III) applications to real-world problems. The chapters cover progressively more advanced topics, with early chapters providing strong foundations on which the later chapters build. Part I describes the nature of spatial datasets in R and methods for manipulating them. It also covers geographic data import/export and transforming coordinate reference systems. Part II represents methods that build on these foundations. It covers advanced map making (including web mapping), "bridges" to GIS, sharing reproducible code, and how to do cross-validation in the presence of spatial autocorrelation. Part III applies the knowledge gained to tackle real-world problems, including representing and modeling transport systems, finding optimal locations for stores or services, and ecological modeling. Exercises at the end of each chapter give you the skills needed to tackle a range of geospatial problems. Solutions for each chapter and supplementary materials providing extended examples are available at <https://geocompr.github.io/geocompkg/articles/>. Dr. Robin Lovelace is a University Academic Fellow at the University of Leeds, where he has taught R for geographic research over many years, with a focus on transport systems. Dr. Jakub Nowosad is an Assistant Professor in the Department of Geoinformation at the Adam Mickiewicz University in Poznan, where his focus is on the analysis of large datasets to understand environmental processes. Dr. Jannes Muenchow is a Postdoctoral Researcher in the GIScience Department at the University of Jena, where he develops and teaches a range of geographic methods, with a focus on ecological modeling, statistical geocomputing, and predictive mapping. All three are active developers and work on a number of R packages, including *stplanr*, *sabre*, and *RQGIS*.

Functional Aesthetics is a sequel to Seymour's highly acclaimed book *"Fashionable Technology"* (Springer 2008) and contains new state-of-the-art and revealing artistic and design examples focusing on the aesthetic and functional aspects. Chapters like *Contextual Prerequisite*, *Body Sculpture*, or *Transparent Sustainability* provide in-depth studies of often visionary projects seen as stimulation for new developments in the matured field of *"Fashionable Technology"*. The book presents inspiring projects between the poles of fashion, design, technology, and sciences. It includes a list of relevant information on DIY resources, publications, inspirations, etc.

Processing is a free, beginner-friendly programming language designed to help non-programmers create interactive art with code. The *SparkFun Guide to Processing*, the first in the *SparkFun Electronics* series, will show you how to craft digital artwork and even combine that artwork with hardware so that it reacts to the world around you. Start with the basics of programming and animation as you draw colorful shapes and make them bounce around the screen. Then move on to a series of hands-on, step-by-step projects that will show you how to: –Make detailed pixel art and scale it to epic proportions –Write a maze game and build a *MaKey* *MaKey* controller with fruit buttons –Play, record, and sample audio to create your own soundboard –Fetch weather data from the Web and build a custom weather dashboard –Create

visualizations that change based on sound, light, and temperature readings With a little imagination and Processing as your paintbrush, you'll be on your way to coding your own gallery of digital art in no time! Put on your artist's hat, and begin your DIY journey by learning some basic programming and making your first masterpiece with The SparkFun Guide to Processing. The code in this book is compatible with Processing 2 and Processing 3.

Today's designers are creating compelling atmospheres and interactive experiences by merging hardware and software with architecture and design. This book is a collection of this innovative work produced where virtual realms meet the real world and where dataflow confronts the human senses. It presents an international spectrum of interdisciplinary projects at the intersection of laboratory, trade show, and urban space that play with the new frontiers of perception, interaction, and staging created by current technology. The work reveals how technology is fundamentally changing and expanding strategies for the targeted use of architecture, art, communication, and design for the future.

Contingent Computation

Art, Computation And Design

An Approach to Imagination, Computation, and Expression

Postdigital Aesthetics

Design by Numbers

Computational Solutions for Knowledge, Art, and Entertainment: Information Exchange Beyond Text

Contagious Architecture

An essential guide for teaching and learning computational art and design: exercises, assignments, interviews, and more than 170 illustrations of creative work. This book is an essential resource for art educators and practitioners who want to explore code as a creative medium, and serves as a guide for computer scientists transitioning from STEM to STEAM in their syllabi or practice. It provides a collection of classic creative coding prompts and assignments, accompanied by annotated examples of both classic and contemporary projects, and more than 170 illustrations of creative work, and features a set of interviews with leading educators. Picking up where standard programming guides leave off, the authors highlight alternative programming pedagogies suitable for the art- and design-oriented classroom, including teaching approaches, resources, and community support structures.

This book gives clear and effective instructions, stuffed with practical examples, to build your own fun, stunning and highly-interactive openFrameworks applications. Each chapter is focused differently and has a new theme to it, This book targets visual artists, designers, programmers and those interested in creative coding by getting started with openFrameworks. This book will help you understand the capabilities of openFrameworks to help you create visually stunning and fully interactive applications. You should have a basic knowledge of object oriented programming, such as C++, Java, Python, ActionScript 3, etc.

The creator of the designer website, maeda@media, explores the computer as an artistic medium, recounting how his students and he have rendered some of the most digitally sophisticated pieces of design in modern history, in a compilation that showcases some of the ACG's key achievements in the fields of digital typography, interaction design, education, and more. Original.

"John Maeda deconstructs the digital world with the earned authority of an M.I.T.-trained computer scientist and a card-carrying artist. Being ambidextrous with Eastern and Western cultures, he can see things most of us overlook. The result is a humor and expression that brings out the best in computers and art."--Nicholas Negroponte John Maeda is one of the world's leading experimental graphic designers and is quickly becoming a digital culture icon. His early preoccupation with the intersection of computer programming and digital art has resulted in a fascinating, interactive, and stunningly beautiful collection of work. Maeda has pioneered many of the key expressive elements that are prevalent on the web today. Among his most well-known works are The Reactive Square, which features a simple black square on a computer screen that changes shape if one yells at it, and Time Paint, in which paint flies across the screen. He has created innovative, interactive calendars, digital services, and advertisements for companies such as Sony, Shiseido, and Absolut Vodka. This is the first publication to present a complete overview of Maeda's work and philosophy. A glorious visual exploration of ideas and graphic form, Maeda @ Media takes you through Maeda's beginnings in early computerized printouts, to his reactive graphics on CD-ROM, to his dynamic experiments on the web, to his pedagogical approach to digital visual art, and finally to his overarching quest to understand the very nature of the relationship between technology and creativity. Six thematic chapters provide an overview of his entire career and research. But this is not just a catalog of older work: interspersed between each chapter is a new visual essay that has been created exclusively for this publication to underline each of the major themes. Coming together in a massive 480 pages, printed in a

dazzling array of color combinations on three different kinds of paper, the result is a manifesto, a finely crafted manual and inspiration sourcebook all in one. With over 1000 illustrations.

The Ethics and Aesthetics of Hacking

Functional Aesthetics

Coding Freedom

The Troubled History of Computer Art

Mastering openFrameworks: Creative Coding Demystified

Soft Innovation

Nietzsche on Art and Life

The chapters in this volume reflect the debates that progressed during the 4th Global Conference on Visions of Humanity in Cyberculture, Cyberspace and Science Fiction, held as a part of Cyber Hub activity in the frames of the ID.net Critical Issues research in Oxford, United Kingdom in July 2009.

A pioneering graphic designer shows how to use the computer as an artistic medium in its own right. Most art and technology projects pair artists with engineers or scientists: the artist has the conception, and the technical person provides the know-how. John Maeda is an artist and a computer scientist, and he views the computer not as a substitute for brush and paint but as an artistic medium in its own right. Design By Numbers is a reader-friendly tutorial on both the philosophy and nuts-and-bolts techniques of programming for artists. Practicing what he preaches, Maeda composed Design By Numbers using a computational process he developed specifically for the book. He introduces a programming language and development environment, available on the Web, which can be freely downloaded or run directly within any JAVA-enabled Web browser.

Appropriately, the new language is called DBN (for "design by numbers"). Designed for "visual" people—artists, designers, anyone who likes to pick up a pencil and doodle—DBN has very few commands and consists of elements resembling those of many other languages, such as LISP, LOGO, C/JAVA, and BASIC. Throughout the book, Maeda emphasizes the importance—and delights—of understanding the motivation behind computer programming, as well as the many wonders that emerge from well-written programs. Sympathetic to the "mathematically challenged," he places minimal emphasis on mathematics in the first half of the book. Because computation is inherently mathematical, the book's second half uses intermediate mathematical concepts that generally do not go beyond high-school algebra. The reader who masters the skills so clearly set out by Maeda will be ready to exploit the true character of digital media design.

As interactive application software such as apps, installations, and multimedia presentations have become pervasive in everyday life, more and more computer scientists, engineers, and technology experts acknowledge the influence that exists beyond visual explanations. Computational Solutions for Knowledge, Art, and Entertainment: Information Exchange Beyond Text focuses on the methods of depicting knowledge-based concepts in order to assert power beyond a visual explanation of scientific and computational notions. This book combines formal descriptions with graphical presentations and encourages readers to interact by creating visual solutions for science-related concepts and presenting data. This reference is essential for researchers, computer scientists, and academics focusing on the integration of science, technology, computing, art, and mathematics for visual problem solving.

First Processing book on the market Processing is a nascent technology rapidly increasing in popularity Links with the creators of Processing will help sell the book

Generative Art

The New Aesthetic and Art

Computers and Creativity

New Tendencies and Bit International, 1961-1973

Interactive Installations and Experiences

10 PRINT CHR\$(205.5+RND(1)); : GOTO 10

A Lexicon

This interdisciplinary volume introduces new theories and ideas on creativity from the perspectives of science and art. Featuring contributions from leading researchers, theorists and artists working in artificial intelligence, generative art, creative computing, music composition, and cybernetics, the book examines the relationship between computation and creativity from both analytic and practical perspectives. Each contributor describes innovative new ways creativity can be understood through, and inspired by, computers. The book tackles critical philosophical questions and discusses the major issues raised by computational creativity, including: whether a computer can exhibit creativity independently of its creator; what kinds of creativity are possible in light of our knowledge from computational simulation, artificial intelligence, evolutionary theory and information theory; and whether we can begin to automate the evaluation of aesthetics and creativity in silico. These important, often controversial questions are contextualised by current thinking in computational creative arts practice. Leading artistic practitioners discuss their approaches to working creatively with computational systems in a diverse array of media, including music, sound art, visual art, and interactivity. The volume also includes a comprehensive review of computational aesthetic evaluation and judgement research, alongside discussion and insights from pioneering artists working with computation as a creative medium over the last fifty years. A distinguishing feature of this volume is that it explains and grounds new theoretical ideas on creativity through practical applications and creative practice. Computers and Creativity will appeal to theorists, researchers in artificial intelligence, generative and evolutionary computing, practicing artists and musicians, students and any reader generally interested in understanding how computers can impact upon creativity. It bridges concepts from computer science, psychology, neuroscience, visual art, music and philosophy in an accessible way, illustrating how computers are fundamentally changing what we can imagine and create, and how we might shape the creativity of the future. Computers and Creativity will appeal to theorists, researchers in artificial intelligence, generative and evolutionary computing, practicing artists and musicians, students and any reader generally interested in understanding how computers can impact upon creativity. It bridges concepts from computer science, psychology, neuroscience, visual art, music and philosophy in an accessible way, illustrating how computers are fundamentally changing what we can imagine and create, and how we might shape the creativity of the future.

A single line of code offers a way to understand the cultural context of computing. This book takes a single line of code—the extremely concise BASIC program for the Commodore 64 inscribed in the title—and uses it as a lens through

which to consider the phenomenon of creative computing and the way computer programs exist in culture. The authors of this collaboratively written book treat code not as merely functional but as a text—in the case of 10 PRINT, a text that appeared in many different printed sources—that yields a story about its making, its purpose, its assumptions, and more. They consider randomness and regularity in computing and art, the maze in culture, the popular BASIC programming language, and the highly influential Commodore 64 computer.

A phenomenological investigation into new media artwork and its relationship to history What does it mean to live in an era of emerging digital technologies? Are computers really as antihistorical as they often seem? Drawing on phenomenology ' s investigation of time and history, Sensations of History uses encounters with new media art to inject more life into these questions, making profound contributions to our understanding of the digital age in the larger scope of history. Sensations of History combines close textual analysis of experimental new media artworks with in-depth discussions of key texts from the philosophical tradition of phenomenology. Through this inquiry, author James J. Hodge argues for the immense significance of new media art in examining just what historical experience means in a digital age. His beautiful, aphoristic style demystifies complex theories and ideas, making perplexing issues feel both graspable and intimate. Highlighting underappreciated, vibrant work in the fields of digital art and video, Sensations of History explores artists like Paul Chan, Phil Solomon, John F. Simon, and Barbara Lattanzi. Hodge ' s provocative interpretations, which bring these artists into dialogue with well-known works, are perfect for scholars of cinema, media studies, art history, and literary studies. Ultimately, Sensations of History presents the compelling case that we are not witnessing the end of history—we are instead seeing its rejuvenation in a surprising variety of new media art.

This book explores computation as a medium for drawing. Exercises, essays, algorithms, diagrams, and drawings are woven together to offer instruction, insight, and theories that are valuable to practicing architects, artists, and scholars.

The SparkFun Guide to Processing

A Touch of Code

Defining Art, Creating the Canon

Computation, Aesthetics, and Space

A Handbook of Software Studies

A Handbook for Computational Art and Design

How Computer Programming Is Changing Writing

Who are computer hackers? What is free software? And what does the emergence of a community dedicated to the production of free and open source software--and to hacking as a technical, aesthetic, and moral project--reveal about the values of contemporary liberalism? Exploring the rise and political significance of the free and open source software (F/OSS) movement in the United States and Europe, Coding Freedom details the ethics behind hackers' devotion to F/OSS, the social codes that guide its production, and the political struggles through which hackers question the scope and direction of copyright and patent law. In telling the story of the F/OSS movement, the book unfolds a broader narrative involving computing, the politics of access, and intellectual property. E. Gabriella Coleman tracks the ways in which hackers collaborate and examines passionate manifestos, hacker humor, free software project governance, and festive hacker conferences. Looking at the ways that hackers sustain their productive freedom, Coleman shows that these activists, driven by a commitment to their work, reformulate key ideals including free speech, transparency, and meritocracy, and refuse restrictive intellectual protections. Coleman demonstrates how hacking, so often marginalized or misunderstood, sheds light on the continuing relevance of liberalism in online collaboration.

"This book outlines a musical journey through Scratch. Scratch is an approachable computer programming environment that contains a rich set of media features, such as music and sound - both of which are explored here. The book features a series of independent musical projects built in Scratch and guides readers through the processes required to create each project. Readers will encounter coding techniques and algorithmic music processes while completing the exercises. In general, the projects are very interactive and encourage readers to make music through playing and composing with each task"--

At its heart this book is about innovation and the innovation process. On the way, it considers aesthetics, design, creativity and the creative industries, and a number of other similar topics. Much of the existing economic literature on innovation has taken a particularly technological or functional viewpoint as to what sort of new products and processes are to be considered innovations. One of the key things this book shows is that there is a type of innovation, here labelled 'soft innovation', primarily concerned with changes in products (and perhaps processes) of an aesthetic or intellectual nature, that has largely been ignored in the study of innovation prevalent in economics. Examples of innovations that, as a result of this refocusing, are here placed at the centre of the analysis include: the writing and publishing of a new book, the writing, production, and launching of a new movie, the development and launch of a new advertising promotion, the design and production of a new range of furniture, and architectural activity in the generation of new built form designs. The realisation of the existence of soft innovation means that, not only is innovation more widespread than previously considered, but that it may also take a different form than commonly considered. Soft Innovation addresses key issues such as: * The measurement of the rate and extent of soft innovation, * The determinants of the rate and direction of soft innovation and its diffusion, * The impacts of soft innovation and diffusion upon outputs, productivity, employment, firm performance, trade, and economic welfare, * Policy, considering whether there is a rationale for government intervention in the soft innovation generation and diffusion processes, and if so what instruments can be used in such intervention? Soft Innovation breaks new ground in the study of innovation, and will be key reading for academics and researchers of Innovation, Marketing, and Design, as well as consultants, practitioners, and policy-makers concerned with the creative industries.

An argument that great expressive power of computational media arises from the construction of phantasms—blends of cultural ideas and sensory imagination. In Phantasmal Media, D. Fox Harrell considers the expressive power of computational media. He argues, forcefully and persuasively, that the great expressive potential of computational media comes from the ability to construct and reveal phantasms—blends of cultural ideas and sensory imagination. These ubiquitous and often-unseen phantasms—cognitive phenomena that include sense of self, metaphors, social categories, narrative, and poetic thinking—influence almost all our everyday experiences. Harrell offers an approach for understanding and designing computational systems that have the power to evoke these phantasms, paying special attention to the exposure of oppressive phantasms and the creation of empowering ones. He argues for the importance of cultural content, diverse worldviews, and social values in computing. The

expressive power of phantasms is not purely aesthetic, he contends; phantasmal media can express and construct the types of meaning central to the human condition. Harrell discusses, among other topics, the phantasm as an orienting perspective for developers; expressive epistemologies, or data structures based on subjective human worldviews; morphic semiotics (building on the computer scientist Joseph Goguen's theory of algebraic semiotics); cultural phantasms that influence consensus and reveal other perspectives; computing systems based on cultural models; interaction and expression; and the ways that real-world information is mapped onto, and instantiated by, computational data structures. The concept of phantasmal media, Harrell argues, offers new possibilities for using the computer to understand and improve the human condition through the human capacity to imagine.

The Laws of Simplicity

Aesthetic Programming

The Art of Coding

Maeda @ Media

Sensations of History

New Essays

Coding Art

How the theoretical tools of literacy help us understand programming in its historical, social and conceptual contexts. The message from educators, the tech community, and even politicians is clear: everyone should learn to code. To emphasize the universality and importance of computer programming, promoters of coding for everyone often invoke the concept of “ literacy, ” drawing parallels between reading and writing code and reading and writing text. In this book, Annette Vee examines the coding-as-literacy analogy and argues that it can be an apt rhetorical frame. The theoretical tools of literacy help us understand programming beyond a technical level, and in its historical, social, and conceptual contexts. Viewing programming from the perspective of literacy and literacy from the perspective of programming, she argues, shifts our understandings of both. Computer programming becomes part of an array of communication skills important in everyday life, and literacy, augmented by programming, becomes more capacious. Vee examines the ways that programming is linked with literacy in coding literacy campaigns, considering the ideologies that accompany this coupling, and she looks at how both writing and programming encode and distribute information. She explores historical parallels between writing and programming, using the evolution of mass textual literacy to shed light on the trajectory of code from military and government infrastructure to large-scale businesses to personal use. Writing and coding were institutionalized, domesticated, and then established as a basis for literacy. Just as societies demonstrated a “ literate mentality ” regardless of the literate status of individuals, Vee argues, a “ computational mentality ” is now emerging even though coding is still a specialized skill.

An argument that we must read code for more than what it does—we must consider what it means. Computer source code has become part of popular discourse. Code is read not only by programmers but by lawyers, artists, pundits, reporters, political activists, and literary scholars; it is used in political debate, works of art, popular entertainment, and historical accounts. In this book, Mark Marino argues that code means more than merely what it does; we must also consider what it means. We need to learn to read code critically. Marino presents a series of case studies—ranging from the Climategate scandal to a hactivist art project on the US-Mexico border—as lessons in critical code reading. Marino shows how, in the process of its circulation, the meaning of code changes beyond its functional role to include connotations and implications, opening it up to interpretation and inference—and misinterpretation and reappropriation. The Climategate controversy, for example, stemmed from a misreading of a bit of placeholder code as a “ smoking gun ” that supposedly proved fabrication of climate data. A poetry generator created by Nick Montfort was remixed and reimagined by other poets, and subject to literary interpretation. Each case study begins by presenting a small and self-contained passage of code—by coders as disparate as programming pioneer Grace Hopper and philosopher Friedrich Kittler—and an accessible explanation of its context and functioning. Marino then explores its extra-functional significance, demonstrating a variety of interpretive approaches.

This collection of short expository, critical and speculative texts offers a field guide to the cultural, political, social and aesthetic impact of software. Experts from a range of disciplines each take a key topic in software and the understanding of software, such as algorithms and logical structures.

Postdigital Aesthetics is a contribution to questions raised by our newly computational everyday lives and the aesthetics which reflect both the postdigital nature of this age, but also critical perspectives of a post-internet world.

Economics, Product Aesthetics, and the Creative Industries

Coding Literacy

Geocomputation with R

Visions in Fashionable Technology

Creative Coding and Computational Art

Software Studies

A practical guide using Processing

Summary Generative Art presents both the technique and the beauty of algorithmic art. The book includes high-quality examples of generative art, along with the specific programmatic steps author and artist Matt Pearson followed to create each unique piece using the Processing programming language. About the Technology Artists have always explored new media, and computer-based artists are no exception. Generative art, a technique where the artist creates print or onscreen images by using computer algorithms, finds the artistic intersection of programming, computer graphics, and individual expression. The book includes a tutorial on Processing, an open source programming language and environment for people who want to create images, animations, and interactions. About the Book Generative Art presents both the techniques and the beauty of algorithmic art. In it, you'll find dozens of high-quality examples of generative art, along with the specific steps the author followed to create

each unique piece using the Processing programming language. The book includes concise tutorials for each of the technical components required to create the book's images, and it offers countless suggestions for how you can combine and reuse the various techniques to create your own works. Purchase of the print book comes with an offer of a free PDF, ePub, and Kindle eBook from Manning. Also available is all code from the book. What's Inside The principles of algorithmic art A Processing language tutorial Using organic, pseudo-random, emergent, and fractal processes

=====
Table of Contents Part 1 Creative Coding Generative Art: In Theory and Practice Processing: A Programming Language for Artists Part 2 Randomness and Noise The Wrong Way to Draw A Line The Wrong Way to Draw a Circle Adding Dimensions Part 3 Complexity Emergence Autonomy Fractals
What is art; why should we value it; and what allows us to say that one work is better than another? Traditional answers have emphasized aesthetic form. But this has been challenged by institutional definitions of art and postmodern critique. The idea of distinctively artistic value based on aesthetic criteria is at best doubted, and at worst, rejected. This book, however, champions these notions in a new way. It does so through a rethink of the mimetic definition of art on the basis of factors which traditional answers neglect, namely the conceptual link between art's aesthetic value and 'non-exhibited' epistemological and historical relations. These factors converge on an expanded notion of the artistic image (a notion which can even encompass music, abstract art, and some conceptual idioms). The image's style serves to interpret its subject-matter. If this style is original (in comparative historical terms) it can manifest that special kind of aesthetic unity which we call art. Appreciation of this involves a heightened interaction of capacities (such as imagination and understanding) which are basic to knowledge and personal identity. By negotiating these factors, it is possible to define art and its canonic dimensions objectively, and to show that aforementioned sceptical alternatives are incomplete and self-contradictory.

When Zagreb was the epicenter of explorations into the aesthetic potential of the new “ thinking machines. ” This book documents a short but intense artistic experiment that took place in Yugoslavia fifty years ago but has been influential far beyond that time and place: the “ little-known story ” of the advent of computers in art. It was through the activities of the New Tendencies movement, begun in Zagreb in 1961, and its supporting institution the Galerija suvremene umjetnosti that the “ thinking machine ” was adopted as an artistic tool and medium. Pursuing the idea of “ art as visual research, ” the New Tendencies movement proceeded along a path that led from Concrete and Constructivist art, Op art, and Kinetic art to computer-generated graphics, film, and sculpture. With their exhibitions and conferences and the 1968 launch of the multilingual, groundbreaking magazine Bit International, the New Tendencies transformed Zagreb—already one of the most vibrant artistic centers in Yugoslavia—into an international meeting place where artists, engineers, and scientists from both sides of the Iron Curtain gathered around the then-new technology. For a brief moment in time, Zagreb was the epicenter of explorations of the aesthetic, scientific, and political potential of the computer. This volume documents that exhilarating period. It includes new essays by Jerko Denegri, Darko Fritz, Margit Rosen, and Peter Weibel; many texts that were first published in New Tendencies exhibition catalogs and Bit International magazine; and historic documents. More than 650 black-and-white and color illustrations testify to the astonishing diversity of the exhibited artworks and introduce the movement's protagonists. Many of the historic photographs, translations, and documents are published here for the first time. Taken together, the images and texts offer the long overdue history of the New Tendencies experiment and its impact on the art of the twentieth century.

As the title suggests, this book explores the concepts of drawing, graphics and animation in the context of coding. In this endeavour, in addition to initiating the process with some historical perspectives on programming languages, it prides itself by presenting complex concepts in an easy-to-understand fashion for students, artists, hobbyists as well as those interested in computer science, computer graphics, digital media, or interdisciplinary studies. Being able to code requires abstract thinking, mathematics skills, spatial ability, logical thinking, imagination, and creativity. All these abilities can be acquired with practice, and can be mastered by practical exposure to art, music, and literature. This book discusses art, poetry and other forms of writing while pondering difficult concepts in programming; it looks at how we use our senses in the process of learning computing and programming. Features: - Introduces coding in a visual way - Explores the elegance behind coding and the outcome - Includes types of outcomes and options for coding - Covers the transition from front-of-classroom instruction to the use of online-streamed video tutorials - Encourages abstract and cognitive thinking, as well as creativity The Art of Coding contains a collection of learning projects for students, instructors and teachers to select specific themes from. Problems and projects are aimed at making the learning process entertaining, while also involving social exchange and sharing. This process allows for programming to become interdisciplinary, enabling projects to be co-developed by specialists from different backgrounds, enriching the value of coding and what it can achieve. The authors of this book hail from three different continents, and have several decades of combined experience in academia, education, science and visual arts.

Computational Drawing

Code as Creative Medium

Artistic Value in an Era of Doubt

Information Exchange Beyond Text

The Language of Drawing, Graphics, and Animation

Create Interactive Art with Code

Scratch Music Projects

Creativity pervades human life. It is the mark of individuality, the vehicle of self-expression, and the engine of progress in every human endeavor. It also raises a wealth of neglected and yet evocative philosophical questions. The Philosophy of Creativity takes up these questions and, in doing so, illustrates the value of interdisciplinary exchange.

The application of the theory and practice of art to computer science: how aesthetics and art can play a role in computing disciplines.

A proposal that algorithms are not simply instructions to be performed but thinking entities that construct digital spatio-temporalities. In Contagious Architecture, Luciana Parisi offers a philosophical inquiry into the

status of the algorithm in architectural and interaction design. Her thesis is that algorithmic computation is not simply an abstract mathematical tool but constitutes a mode of thought in its own right, in that its operation extends into forms of abstraction that lie beyond direct human cognition and control. These include modes of infinity, contingency, and indeterminacy, as well as incomputable quantities underlying the iterative process of algorithmic processing. The main philosophical source for the project is Alfred North Whitehead, whose process philosophy is specifically designed to provide a vocabulary for “ modes of thought ” exhibiting various degrees of autonomy from human agency even as they are mobilized by it. Because algorithmic processing lies at the heart of the design practices now reshaping our world—from the physical spaces of our built environment to the networked spaces of digital culture—the nature of algorithmic thought is a topic of pressing importance that reraises questions of control and, ultimately, power. Contagious Architecture revisits cybernetic theories of control and information theory's notion of the incomputable in light of this rethinking of the role of algorithmic thought. Informed by recent debates in political and cultural theory around the changing landscape of power, it links the nature of abstraction to a new theory of power adequate to the complexities of the digital world.

Abstraction, Experience, and Indeterminacy in Computational Aesthetics