The Cs Detective An Algorithmic Tale Of Crime Conspiracy And Computation

Impractical Python ProjectsSyntactic StructuresThe Art of Game DesignCoding with MinecraftAlgorithms to Live BySchemeHow Not to be WrongThe Scheme Programming LanguageMathematics and ComputationThe Top Ten Algorithms in Data MiningThink JavaIntroduction to C Programming :Think JuliaSearch AlgorithmsData Structures and Abstractions with JavaStoner PizzaIntroduction to ProbabilityDive Into AlgorithmsHands-On Design Patterns and Best Practices with JuliaActual CausalityNetopiaBest Practices of Spell DesignLaunchCosmo to the Rescue (My Pet Slime Book 2)Machine LearningHandbook of Research on Tools for Teaching Computational Thinking in P-12 EducationOnce Upon an AlgorithmBad ChoicesBayesian Statistics the Fun WayThe Wisdom of CrowdsQuantum Computer ScienceFractal Art Adult Coloring BookComputational Fairy TalesFerocious LogicsMy Pet Slime (My Pet Slime Book 1)Data Structures: Abstraction and Design Using Java, 3rd EditionLauren IpsumThe CS DetectiveProgramming PearlsGamer Bandit

Impractical Python Projects

While the growth of computational thinking has brought new awareness to the importance of computing education, it has also created new challenges. Many educational initiatives focus solely on the programming aspects, such as variables, loops, conditionals, parallelism, operators, and data handling, divorcing computing from real-world contexts and applications. This decontextualization threatens to make learners believe that they do not need to learn computing, as they cannot envision a future in which they will need to use it, just as many see math and physics education as unnecessary. The Handbook of Research on Tools for Teaching Computational Thinking in P-12 Education is a cutting-edge research publication that examines the implementation of computational thinking into school curriculum in order to develop creative problem-solving skills and to build a computational identity which will allow for future STEM growth. Moreover, the book advocates for a new approach to computing education that argues that while learning about computing, young people should also have opportunities to create with computing, which will have a direct impact on their lives and their communities. Featuring a wide range of topics such as assessment, digital teaching, and educational robotics, this book is ideal for academicians, instructional designers, teachers, education professionals, administrators, researchers, and students.

Syntactic Structures

Identifying some of the most influential algorithms that are widely used in the data mining community, The Top Ten Algorithms in Data Mining provides a description of each algorithm, discusses its impact, and reviews current and future research. Thoroughly evaluated by independent reviewers, each chapter focuses on a particular algorithm and is written by either the original authors of the algorithm or world-class researchers who have extensively studied the respective algorithm. The book concentrates on the following important algorithms: C4.5, k-Means, SVM, Apriori, EM, PageRank, AdaBoost, kNN, Naive Bayes, and CART. Examples illustrate how each algorithm works and highlight its overall performance in a real-world application. The text covers key topics—including classification, clustering, statistical learning, association analysis, and link mining—in data mining research and development as well as in data mining, machine learning, and artificial intelligence courses. By naming the leading algorithms in this field, this book encourages the use of data mining techniques in a broader realm of real-world applications. It should inspire more data mining researchers to further explore the impact and novel research issues of these algorithms.

The Art of Game Design

Tommy, Rafeal, Edward and Grant---servers at Stoner Pizza---are economically, psychologically, and socially trapped in a life maze they didn't create nor understand. They're second generation high school and college dropouts immersed in an unrelenting algorithm-driven computer-technology in a society cascading in a downward spiral of dumbing down and continued compromise. Pizza Stoners like Tommy, Rafael, Edward and Grant live at home with their parents, never leave their room except for the most minimal of part-time jobs---antisocials lost in a blizzard of computers, TV, Internet technology, entertainment and porn all mixed together. The Pizza Stoner, like his Japanese cousin, the Hikikomori, is the product of decades of economic recession, stagnation and despair--- the wasted generation of Japan--- not educated, unemployed, not trained or trainable in a society going nowhere. The symbol of this class is a Harvard PhD who waits on tables at a fast food restaurant with dim prospects for the foreseeable future. Relevant to American society today, Stoner Pizza is a tragic, reflective, satirical and thoughtful look at lost dreams and opportunities and settling for less in the affluent country of America today.

Coding with Minecraft

Algorithms to Live By

Twenty five highly detailed grayscale coloring pages feature the fantasy fractal artwork of Tabz Jones. Fractal art is a definite product of the modern age. Where else could you use mathmatical algorithms to create images? The repeating patterns of the equations are manipulated using a software program to change color, form, and shape to create a final artwork that is a one of a kind.

Scheme

Contemporary power manifests in the algorithmic. And yet this power seems incomprehensible: understood as code, it becomes apolitical; understood as a totality, it becomes overwhelming. This book takes an alternate approach, using it to unravel the operations of Uber and Palantir, Airbnb and Amazon Alexa. Moving off the whiteboard and into the world, the algorithmic must negotiate with frictions-the 'merely' technical routines of distributing data and running tasks coming together into broader social forces that shape subjectivities, steer bodies, and calibrate relationships. Driven by the imperatives of capital, the algorithmic exhausts subjects and spaces, a double move seeking to both exhaustively

apprehend them and exhaust away their productivities. But these on-the-ground encounters also reveal that force is never guaranteed. The irreducibility of the world renders logic inadequate and control gives way to contingency. This work was published by Saint Philip Street Press pursuant to a Creative Commons license permitting commercial use. All rights not granted by the work's license are retained by the author or authors.

How Not to be Wrong

Developed from celebrated Harvard statistics lectures, Introduction to Probability provides essential language and tools for understanding statistics, randomness, and uncertainty. The book explores a wide variety of applications and examples, ranging from coincidences and paradoxes to Google PageRank and Markov chain Monte Carlo (MCMC). Additional

The Scheme Programming Language

"The Best Practices of Spell Design introduces practical aspects of software development that are often learned through painful experience. Through Marcus and Shelly's quest, the story encourages readers to think about how to write readable, well-tested and maintainable programs."--P. [4] of

cover.

Mathematics and Computation

"Using the mathematician's method of analyzing life and exposing the hard-won insights of the academic community to the layman, minus the jargon Ellenberg pulls from history as well as from the latest theoretical developments to provide those not trained in math with the knowledge they need"--

The Top Ten Algorithms in Data Mining

Design and develop high-performance, reusable, and maintainable applications using traditional and modern Julia patterns with this comprehensive guide Key Features Explore useful design patterns along with object-oriented programming in Julia 1.0 Implement macros and metaprogramming techniques to make your code faster, concise, and efficient Develop the skills necessary to implement design patterns for creating robust and maintainable applications Book Description Design patterns are fundamental techniques for developing reusable and maintainable code. They provide a set of proven solutions that allow developers to solve problems in software development quickly. This book will demonstrate how to leverage design patterns with real-world applications. Starting with

an overview of design patterns and best practices in application design, you'll learn about some of the most fundamental Julia features such as modules, data types, functions/interfaces, and metaprogramming. You'll then get to grips with the modern Julia design patterns for building large-scale applications with a focus on performance, reusability, robustness, and maintainability. The book also covers anti-patterns and how to avoid common mistakes and pitfalls in development. You'll see how traditional object-oriented patterns can be implemented differently and more effectively in Julia. Finally, you'll explore various use cases and examples, such as how expert Julia developers use design patterns in their open source packages. By the end of this Julia programming book, you'll have learned methods to improve software design, extensibility, and reusability, and be able to use design patterns efficiently to overcome common challenges in software development. What you will learn Master the Julia language features that are key to developing large-scale software applications Discover design patterns to improve overall application architecture and design Develop reusable programs that are modular, extendable, performant, and easy to maintain Weigh up the pros and cons of using different design patterns for use cases Explore methods for transitioning from object-oriented programming to using equivalent or more

advanced Julia techniques Who this book is for This book is for beginner to intermediate-level Julia programmers who want to enhance their skills in designing and developing large-scale applications.

Think Java

Data Structures: Abstraction and Design Using Java, 3rd Edition, combines a strong emphasis on problem solving and software design with the study of data structures. The authors discuss applications of each data structure to motivate its study. After providing the specification (interface) and the implementation (a Java class), case studies that use the data structure to solve a significant problem are introduced.

Introduction to C Programming :

Piper Maclane wants a pet so badly, but she's allergic to everything. So she creates her ownâe*out of slime! From Epic! Originals, My Pet Slime is a sweet, slimy series about a girl who makes the most of a challenging situation and is rewarded with a bit of magic! Piper wishes she could have a pet to care for and snuggle with and take to school for Show and Tell. All of the other kids have one. But she's allergic to every animal imaginable! So instead, she makes her own pet out of slime, with big eyes, a little mouth,

and two arms just long enough for hugs. It would be the perfect pet for Piper, if it were real. When a strange turn of events brings her slime to life, Piper learns how fun, challenging, and messy it can be to care for a petâe*especially one made out of slime! At the back of the book, learn about real-life slime and follow the recipe to create a "pet" of your own!

Think Julia

The perfect reality is just a thought away, but it comes at a chilling price. In a near future world, a social network by the name of MINDS gifts its users with the unique ability to communicate directly by thoughts and recreate reality by their deepest fantasies and desires. In the MINDS network, desires are actualized in a spilt of a second and unwanted elements erased from reality. The advanced algorithms of MINDS enable the user to virtually visit any place in the world or in history within seconds, in an utterly realistic three-dimensional reality. Humans can use their mind to determine everything they please - from the weather to the way their friends look like and behave. But the ultimate comfort comes at a price, and the unexpected results reveal the grand plan behind MINDS. What stands behind this magnificent social network, and what are the consequences when the virtual merges with the

actual? Netopia follows the heroes of the network in the days before its establishment, through its launch, relationships and love, and the unexpected change it wreaks on the enthusiastic users who had no idea what awaited them the moment they gave up control over their minds. Is the MINDS network a dream come true or a nightmare come to life? Netopia is beyond science fiction: it is a groundbreaking novel that explores the implications of communication technology on human nature and society, the preference of warmth for realistic animal doll pets but coldness towards humans. the novel speaks to the connected and to the disconnected, to the great minds and the Neverminds. Netopia is a gleaming hope and a dark warning. So, where is your mind? Scroll up to grab your copy now.

Search Algorithms

An introduction to computational complexity theory, its connections and interactions with mathematics, and its central role in the natural and social sciences, technology, and philosophy Mathematics and Computation provides a broad, conceptual overview of computational complexity theory—the mathematical study of efficient computation. With important practical applications to computer science and industry, computational complexity theory has evolved into a highly

interdisciplinary field, with strong links to most mathematical areas and to a growing number of scientific endeavors. Avi Wigderson takes a sweeping survey of complexity theory, emphasizing the field's insights and challenges. He explains the ideas and motivations leading to key models, notions, and results. In particular, he looks at algorithms and complexity, computations and proofs, randomness and interaction, quantum and arithmetic computation, and cryptography and learning, all as parts of a cohesive whole with numerous cross-influences. Wigderson illustrates the immense breadth of the field, its beauty and richness, and its diverse and growing interactions with other areas of mathematics. He ends with a comprehensive look at the theory of computation, its methodology and aspirations, and the unique and fundamental ways in which it has shaped and will further shape science, technology, and society. For further reading, an extensive bibliography is provided for all topics covered. Mathematics and Computation is useful for undergraduate and graduate students in mathematics, computer science, and related fields, as well as researchers and teachers in these fields. Many parts require little background, and serve as an invitation to newcomers seeking an introduction to the theory of computation. Comprehensive coverage of computational complexity theory, and beyond High-level, intuitive exposition, which brings conceptual

clarity to this central and dynamic scientific discipline Historical accounts of the evolution and motivations of central concepts and models A broad view of the theory of computation's influence on science, technology, and society Extensive bibliography

Data Structures and Abstractions with Java

Using the latest features of Java 5, this unique object-oriented presentation introduces readers to data structures via thirty, manageable chapters. KEY FeaturesTOPICS: Introduces each ADT in its own chapter, including examples or applications. Provides aA variety of exercises and projects, plus additional self-assessment questions throughout. the text Includes generic data types as well as enumerations, for-each loops, the interface Iterable, the class Scanner, assert statements, and autoboxing and unboxing. Identifies important Java code as a Listing. Provides NNotes and Pprogramming Ttips in each chapter. For programmers and software engineers interested in learning more about data structures and abstractions.

Stoner Pizza

A fascinating exploration of how insights

from computer algorithms can be applied to our everyday lives, helping to solve common decision-making problems and illuminate the workings of the human mind All our lives are constrained by limited space and time, limits that give rise to a particular set of problems. What should we do, or leave undone, in a day or a lifetime? How much messiness should we accept? What balance of new activities and familiar favorites is the most fulfilling? These may seem like uniquely human quandaries, but they are not: computers, too, face the same constraints, so computer scientists have been grappling with their version of such issues for decades. And the solutions they've found have much to teach us. In a dazzlingly interdisciplinary work, acclaimed author Brian Christian and cognitive scientist Tom Griffiths show how the algorithms used by computers can also untangle very human questions. They explain how to have better hunches and when to leave things to chance, how to deal with overwhelming choices and how best to connect with others. From finding a spouse to finding a parking spot, from organizing one's inbox to understanding the workings of memory, Algorithms to Live By transforms the wisdom of computer science into strategies for human living.

Introduction to Probability

In the 1990's it was realized that quantum physics has some spectacular applications in computer science. This book is a concise introduction to quantum computation, developing the basic elements of this new branch of computational theory without assuming any background in physics. It begins with an introduction to the quantum theory from a computer-science perspective. It illustrates the quantum-computational approach with several elementary examples of quantum speed-up, before moving to the major applications: Shor's factoring algorithm, Grover's search algorithm, and quantum error correction. The book is intended primarily for computer scientists who know nothing about quantum theory, but will also be of interest to physicists who want to learn the theory of quantum computation, and philosophers of science interested in quantum foundational issues. It evolved during six years of teaching the subject to undergraduates and graduate students in computer science, mathematics, engineering, and physics, at Cornell University.

Dive Into Algorithms

"A mystery novel for computer science students and enthusiasts that introduces the concepts behind search algorithms and data structures. Each chapter teaches a new concept, ending with a technical

explanation"--

Hands-On Design Patterns and Best Practices with Julia

In this fascinating book, New Yorker business columnist James Surowiecki explores a deceptively simple idea: Large groups of people are smarter than an elite few, no matter how brilliant—better at solving problems, fostering innovation, coming to wise decisions, even predicting the future. With boundless erudition and in delightfully clear prose, Surowiecki ranges across fields as diverse as popular culture, psychology, ant biology, behavioral economics, artificial intelligence, military history, and politics to show how this simple idea offers important lessons for how we live our lives, select our leaders, run our companies, and think about our world.

Actual Causality

Bayesian Statistics the Fun Way gets you understanding the theory behind data analysis without making you slog through a load of dry concepts first - with no programming experience necessary. You'll learn about probability with LEGO, statistics through Star Wars, distributions with bomb fuses, estimation through precipitation, and come away with some strong mathematical reasoning

skills. This is a super approachable book for people who need to do data science and probability work in their lives, but never got a good grip on the underlying theory.

Netopia

Using ComputerCraft, a free Minecraft mod, and the Lua programming language (used by professional game developers), you'll write programs to make your bots dig mines, chop wood, craft items, and build anything you can imagine. Sit back, relax, and make your turtle army do the work! The book provides the full code needed to write each program and walks you through it line by line, explaining how things work along the way. By book's end, you ll have a handle on computer science basics, a taste of how fun and powerful coding can be, and plenty of turtle robots at your command.

Best Practices of Spell Design

Currently used at many colleges, universities, and high schools, this hands-on introduction to computer science is ideal for people with little or no programming experience. The goal of this concise book is not just to teach you Java, but to help you think like a computer scientist. You'll learn how to program—a useful skill by itself—but you'll also discover how to use programming

as a means to an end. Authors Allen Downey and Chris Mayfield start with the most basic concepts and gradually move into topics that are more complex, such as recursion and object-oriented programming. Each brief chapter covers the material for one week of a college course and includes exercises to help you practice what you've learned. Learn one concept at a time: tackle complex topics in a series of small steps with examples Understand how to formulate problems, think creatively about solutions, and write programs clearly and accurately Determine which development techniques work best for you, and practice the important skill of debugging Learn relationships among input and output, decisions and loops, classes and methods, strings and arrays Work on exercises involving word games, graphics, puzzles, and playing cards

Launch

Basic, no nonsense introduction to the programming language Scheme

Cosmo to the Rescue (My Pet Slime Book 2)

A new approach for defining causality and such related notions as degree of responsibility, degrees of blame, and causal explanation. Causality plays a central role

in the way people structure the world; we constantly seek causal explanations for our observations. But what does it even mean that an event C "actually caused" event E? The problem of defining actual causation goes beyond mere philosophical speculation. For example, in many legal arguments, it is precisely what needs to be established in order to determine responsibility. The philosophy literature has been struggling with the problem of defining causality since Hume. In this book, Joseph Halpern explores actual causality, and such related notions as degree of responsibility, degree of blame, and causal explanation. The goal is to arrive at a definition of causality that matches our natural language usage and is helpful, for example, to a jury deciding a legal case, a programmer looking for the line of code that cause some software to fail, or an economist trying to determine whether austerity caused a subsequent depression. Halpern applies and expands an approach to causality that he and Judea Pearl developed, based on structural equations. He carefully formulates a definition of causality, and building on this, defines degree of responsibility, degree of blame, and causal explanation. He concludes by discussing how these ideas can be applied to such practical problems as accountability and program verification. Technical details are generally confined to the final section of each chapter and can be skipped by non-mathematical readers.

Machine Learning

Scheme's essential features, listed below, are surveyed in this book. * Symbolic expressions are used to represent both code and data. * Functions-or more generally, procedures- are first-class values that can be stored in data structures and passed to and returned from other functions. * Closures provide the convenience of lexical scoping. * Tail recursive algorithms execute with the same efficiency as imperative loops. * Continuations allow Scheme to be extended with user-defined, dynamic flow control mechanisms such as exception handling, co-routines, nondeterministic backtracking, and logic variables, to name a few. * Scheme's hygienic syntax extension mechanism puts user-defined syntax on an equal footing with its own syntax.These essential features make Scheme one of the most expressive, extensible, yet minimal computer languages in existence. The purpose of this book is to expedite the reader's learning of Scheme or to serve as a quick reference.

Handbook of Research on Tools for Teaching Computational Thinking in P-12 Education

How Hansel and Gretel, Sherlock Holmes, the movie Groundhog Day, Harry Potter, and other familiar stories illustrate the concepts of

computing. Picture a computer scientist, staring at a screen and clicking away frantically on a keyboard, hacking into a system, or perhaps developing an app. Now delete that picture. In Once Upon an Algorithm, Martin Erwig explains computation as something that takes place beyond electronic computers, and computer science as the study of systematic problem solving. Erwig points out that many daily activities involve problem solving. Getting up in the morning, for example: You get up, take a shower, get dressed, eat breakfast. This simple daily routine solves a recurring problem through a series of well-defined steps. In computer science, such a routine is called an algorithm. Erwig illustrates a series of concepts in computing with examples from daily life and familiar stories. Hansel and Gretel, for example, execute an algorithm to get home from the forest. The movie Groundhog Day illustrates the problem of unsolvability; Sherlock Holmes manipulates data structures when solving a crime; the magic in Harry Potter's world is understood through types and abstraction; and Indiana Jones demonstrates the complexity of searching. Along the way, Erwig also discusses representations and different ways to organize data; "intractable" problems; language, syntax, and ambiguity; control structures, loops, and the halting problem; different forms of recursion; and rules for finding errors in algorithms. This engaging

book explains computation accessibly and shows its relevance to daily life. Something to think about next time we execute the algorithm of getting up in the morning.

Once Upon an Algorithm

Impractical Python Projects is a collection of fun and educational projects designed to entertain programmers while enhancing their Python skills. It picks up where the complete beginner books leave off, expanding on existing concepts and introducing new tools that you'll use every day. And to keep things interesting, each project includes a zany twist featuring historical incidents, pop culture references, and literary allusions. You'll flex your problem-solving skills and employ Python's many useful libraries to do things like: - Help James Bond crack a high-tech safe with a hill-climbing algorithm - Write haiku poems using Markov Chain Analysis - Use genetic algorithms to breed a race of gigantic rats - Crack the world's most successful military cipher using cryptanalysis - Derive the anagram, "I am Lord Voldemort" using linguistical sieves - Plan your parents' secure retirement with Monte Carlo simulation - Save the sorceress Zatanna from a stabby death using palingrams - Model the Milky Way and calculate our odds of detecting alien civilizations - Help the world's smartest woman win the Monty Hall

problem argument - Reveal Jupiter's Great Red Spot using optical stacking - Save the head of Mary, Queen of Scots with steganography - Foil corporate security with invisible electronic ink Simulate volcanoes, map Mars, and more, all while gaining valuable experience using free modules like Tkinter, matplotlib, Cprofile, Pylint, Pygame, Pillow, and Python-Docx. Whether you're looking to pick up some new Python skills or just need a pick-me-up, you'll find endless educational, geeky fun with Impractical Python Projects.

Bad Choices

Something happens in students when they define themselves as makers and inventors and creators. They discover powerful skills-problem-solving, critical thinking, and imagination-that will help them shape the world's future our future. If that's true, why isn't creativity a priority in more schools today? Educators John Spencer and A.J. Juliani know firsthand the challenges teachers face every day: School can be busy. Materials can be scarce. The creative process can seem confusing. Curriculum requirements can feel limiting. Those challenges too often bully creativity, pushing it to the side as an "enrichment activity" that gets put off or squeezed into the tiniest time block. We can do better. We must do better if we're going to prepare students for their future. LAUNCH:

Using Design Thinking to Boost Creativity and Bring Out the Maker in Every Student provides a process that can be incorporated into every class at every grade level even if you don't consider yourself a "creative teacher." And if you dare to innovate and view creativity as an essential skill, you will empower your students to change the world–starting right now. Look, Listen, and Learn Ask Lots of Questions Understand the Problem or Process Navigate Ideas Create Highlight What's Working and Failing Are you ready to LAUNCH?

Bayesian Statistics the Fun Way

Dive Into Algorithms is a broad introduction to algorithms using the Python Programming Language. Dive Into Algorithms is a wide-ranging, Pythonic tour of many of the world's most interesting algorithms. With little more than a bit of computer programming experience and basic high-school math, you'll explore standard computer science algorithms for searching, sorting, and optimization; human-based algorithms that help us determine how to catch a baseball or eat the right amount at a buffet; and advanced algorithms like ones used in machine learning and artificial intelligence. You'll even explore how ancient Egyptians and Russian peasants used algorithms to multiply numbers, how the ancient Greeks used them to find greatest common divisors, and how Japanese scholars in

the age of samurai designed algorithms capable of generating magic squares. You'll explore algorithms that are useful in pure mathematics and learn how mathematical ideas can improve algorithms. You'll learn about an algorithm for generating continued fractions, one for quick calculations of square roots, and another for generating seemingly random sets of numbers. You'll also learn how to: • Use algorithms to debug code, maximize revenue, schedule tasks, and create decision trees • Measure the efficiency and speed of algorithms • Generate Voronoi diagrams for use in various geometric applications • Use algorithms to build a simple chatbot, win at board games, or solve sudoku puzzles • Write code for gradient ascent and descent algorithms that can find the maxima and minima of functions • Use simulated annealing to perform global optimization • Build a decision tree to predict happiness based on a person's characteristics Once you've finished this book you'll understand how to code and implement important algorithms as well as how to measure and optimize their performance, all while learning the nitty-gritty details of today's most powerful algorithms.

The Wisdom of Crowds

Lauren Ipsum is a whimsical journey through a land where logic and computer science come to life. Meet Lauren, an adventurer lost in

Userland who needs to find her way home by solving a series of puzzles. As she visits places like the Push & Pop Café and makes friends with people like Hugh Rustic and the Wandering Salesman, Lauren learns about computer science without even realizing it—and so do you! Read Lauren Ipsum yourself or with someone littler than you, then flip to the notes at the back of the book to learn more about logic and computer science in the real world. Suggested for ages 10+

Quantum Computer Science

Fractal Art Adult Coloring Book

Have you ever thought that computer science should include more dragons and wizards? Computational Fairy Tales introduces principles of computational thinking, illustrating high-level computer science concepts, the motivation behind them, and their application in a non-computer—fairy tale—domain. It's a quest that will take you from learning the basics of programming in a blacksmith's forge to fighting curses with recursion.Fifteen seers delivered the same prophecy, without so much as a single minstrel to lighten the mood: an unknown darkness threatens the kingdom. Suddenly, Princess Ann finds herself sent forth alone to save the kingdom. Leaving behind her home,

family, and pet turtle Fido, Princess Ann must face goblin attacks, magical curses, arrogant scholars, an unpleasant oracle, and rude Boolean waiters. Along the way she must build a war chest of computational knowledge to survive the coming challenge.

Computational Fairy Tales

Machine Learning for Beginners Your Definitive guide For Neural Networks, Algorithms, Random Forests and Decision Trees Made Simple Have you ever pause and wonder why some companies like Amazon knows what you like or browsing and make timely recommendations to you? Have you notice that some ads are really "sticky" and seem to follow you from your favorite websites to your Facebook or LinkedIn. Everyone hates spam emails and if you have email accounts from different email providers, you can see which is doing a better job in filtering them out for you. Machine Learning has become an important backend pillar for technologies to innovate and improve while continue learning to bring us the best possible results that we desire based on our feedbacks. With time, the results get better and more accurate which bring us better values such as convenience and save us time. In this book, you will have a peek into what Machine Learning is all about and over the vast applications, which underpins this revolutionary AI technology.

What you will learn in Machine Learning: Your Definitive guide For Neural Networks, Algorithms, Random Forests and Decision Trees Made Simple You will learn the in and outs of machine learning algorithms You will discover some of the applications that have been developed as a result of machine learning You will learn an important chapter that is fundamental to applying machine learning You will learn what are the building blocks of machine learning You will explore the 3 different types of machine learning algorithms And many more.. This Machine Learning: Your Definitive guide For Neural Networks, Algorithms, Random Forests and Decision Trees Made Simple is your must have guide to explore and learn about machine learning, neural networks, algorithms (Markov, Bayes, KNN, many more), random forests and decision trees. Download Machine Learning : Your Definitive guide For Neural Networks, Algorithms, Random Forests and Decision Trees Made Simple You don't need a Kindle to read this book! With free delivery right to your device, you can open it on your PC, Tablet or Smartphone with the free Kindle app. The Bottom Line: What are you waiting for? Start today by making the smartest investment you could possibly make. An investment in yourself, your knowledge and your growth. Don't hesitate to pick up your copy today by clicking the BUY NOW button at the top of this page!

Ferocious Logics

In Book 2 of the My Pet Slime series, Piper and Cosmo are in a race against slime! Piper was beyond surprised when her slime magically came to life and became her adorable pet, Cosmo. Now she must deal with an even bigger shocker: Her grandma has gone missing. With the help of their new friend Claire, it's up to Piper and Cosmo to save Grandma Sadie!

My Pet Slime (My Pet Slime Book 1)

The wildly popular author of Bad Arguments returns with a funny, smart introduction to algorithms—those perennially misunderstood, increasingly important problem-solving rules that can save you time and lead to better choices, every day. Why is Facebook so good at predicting what you like? How do you discover new music? What's the best way to sort your laundry? Readers around the world have embraced Ali Almossawi's whimsical illustrations—drawn by his collaborator Alejandro Giraldo—and his funny, clarifying explanations of complex subjects. In fewer than 200 pages, Almossawi demystifies a new topic of increasing relevance to our lives: algorithms. Bad Choices is a book for anyone who's looked at a given task and wondered if there was a better, faster way to get the task done. What's the best way to organize a grocery list? What's the secret to being more

productive at work? How can we better express ourselves in 140-characters? Presenting us with alternative methods for tackling twelve different scenarios, Almossawi guides us to better choices that borrow from same systems that underline a computer word processor, a Google search engine, or a Facebook ad. Once you recognize what makes a method faster and more efficient, you'll become a more nimble, creative problem-solver, ready to face new challenges. Bad Choices will open the world of algorithms to all readers making this a perennial go-to for fans of quirky, accessible science books. From the Hardcover edition.

Data Structures: Abstraction and Design Using Java, 3rd Edition

If you're just learning how to program, Julia is an excellent JIT-compiled, dynamically typed language with a clean syntax. This hands-on guide uses Julia 1.0 to walk you through programming one step at a time, beginning with basic programming concepts before moving on to more advanced capabilities, such as creating new types and multiple dispatch. Designed from the beginning for high performance, Julia is a general-purpose language ideal for not only numerical analysis and computational science but also web programming and scripting. Through exercises in each chapter, you'll try

out programming concepts as you learn them. Think Julia is perfect for students at the high school or college level as well as self-learners and professionals who need to learn programming basics. Start with the basics, including language syntax and semantics Get a clear definition of each programming concept Learn about values, variables, statements, functions, and data structures in a logical progression Discover how to work with files and databases Understand types, methods, and multiple dispatch Use debugging techniques to fix syntax, runtime, and semantic errors Explore interface design and data structures through case studies

Lauren Ipsum

Anyone can master the fundamentals of game design - no technological expertise is necessary. The Art of Game Design: A Book of Lenses shows that the same basic principles of psychology that work for board games, card games and athletic games also are the keys to making top-quality videogames. Good game design happens when you view your game from many different perspectives, or lenses. While touring through the unusual territory that is game design, this book gives the reader one hundred of these lenses - one hundred sets of insightful questions to ask yourself that will help make your game better. These lenses are gathered from fields as diverse as

psychology, architecture, music, visual design, film, software engineering, theme park design, mathematics, writing, puzzle design, and anthropology. Anyone who reads this book will be inspired to become a better game designer – and will understand how to do it.

The CS Detective

Programming Pearls

Essential C Programming Skills-Made Easy-Without Fear! Write powerful C programs…without becoming a technical expert! This book is the fastest way to get comfortable with C, one incredibly clear and easy step at a time. You'll learn all the basics: how to organize programs, store and display data, work with variables, operators, I/O, pointers, arrays, functions, and much more. C programming has neverbeen this simple! This C Programming book gives a good start and complete introduction for C Programming for Beginner's. Learn the all basics and advanced features of C programming in no time from Bestselling Programming Author Harry. H. Chaudhary. This Book, starts with the basics; I promise this book will make you 100% expert level champion of C Programming. This book contains 1000+ Live C Program's code examples, and 500+ Lab

Exercise & 200+ Brain Wash Topic-wise Code book and 20+ Live software Development Project's. All what you need ! Isn't it ? Write powerful C programs…without becoming a technical expert! This book is the fastest way to get comfortable with C, one incredibly clear and easy step at a time. You'll learn all the basics: how to organize programs, store and display data, work with variables, operators, I/O, pointers, arrays, functions, and much more. (See Below List)C programming has never been this simple! Who knew how simple C programming could be? This is today's best beginner's guide to writing C programs-and to learning skills you can use with practically any language. Its simple, practical instructions will help you start creating useful, reliable C code. This book covers common core syllabus for BCA, MCA, B.TECH, BS (CS), MS (CS), BSC-IT (CS), MSC-IT (CS), and Computer Science Professionals as well as for Hackers. This Book is very serious C Programming stuff: A complete introduction to C Language. You'll learn everything from the fundamentals to advanced topics. If you've read this book, you know what to expect a visually rich format designed for the way your brain works. If you haven't, you're in for a treat. You'll see why people say it's unlike any other C book you've ever read. Learning a new language is no easy. You might think the problem is your brain. It seems to have a mind of its own, a mind that doesn't always want to take in the

dry, technical stuff you're forced to study. The fact is your brain craves novelty. It's constantly searching, scanning, waiting for something unusual to happen. After all, that's the way it was built to help you stay alive. It takes all the routine, ordinary, dull stuff and filters it to the background so it won't interfere with your brain's real work--recording things that matter. How does your brain know what matters? (A) 1000+ Live C Program's code examples, (B) 500+ Lab Exercises, (C) 200+ Brain Wash Topic-wise Code (D) 20+ Live software Development Project's. (E) Learn Complete C- without fear, . || Inside Chapters. || 1. Preface - Page-6, || Introduction to C. 2. Elements of C Programming Language. 3. Control statements (conditions). 4. Control statements (Looping). 5. One dimensional Array. 6. Multi-Dimensional Array. 7. String (Character Array). 8. Your Brain on Functions. 9. Your Brain on Pointers. 10. Structure, Union, Enum, Bit Fields, Typedef. 11. Console Input and Output. 12. File Handling In C. 13. Miscellaneous Topics. 14. Storage Class. 15. Algorithms. 16. Unsolved Practical Problems. 17. PART-II-120+ Practical Code Chapter-Wise. 18. Creating & Inserting own functions in Liberary. 19. Graphics Programming In C. 20. Operating System Development -Intro. 21. C Programming Guidelines. 22. Common C Programming Errors. 23. Live Software Development Using C.

Gamer Bandit

When programmers list their favorite books, Jon Bentley's collection of programming pearls is commonly included among the classics. Just as natural pearls grow from grains of sand that irritate oysters, programming pearls have grown from real problems that have irritated real programmers. With origins beyond solid engineering, in the realm of insight and creativity, Bentley's pearls offer unique and clever solutions to those nagging problems. Illustrated by programs designed as much for fun as for instruction, the book is filled with lucid and witty descriptions of practical programming techniques and fundamental design principles. It is not at all surprising that Programming Pearls has been so highly valued by programmers at every level of experience. In this revision, the first in 14 years, Bentley has substantially updated his essays to reflect current programming methods and environments. In addition, there are three new essays on testing, debugging, and timing set representations string problems All the original programs have been rewritten, and an equal amount of new code has been generated. Implementations of all the programs, in C or C++, are now available on the Web. What remains the same in this new edition is Bentley's focus on the hard core of

programming problems and his delivery of workable solutions to those problems. Whether you are new to Bentley's classic or are revisiting his work for some fresh insight, the book is sure to make your own list of favorites.

Read More About The Cs Detective An Algorithmic Tale Of Crime Conspiracy And Computation

Arts & Photography
Biographies & Memoirs
Business & Money
Children's Books
Christian Books & Bibles
Comics & Graphic Novels
Computers & Technology
Cookbooks, Food & Wine
Crafts, Hobbies & Home
Education & Teaching
Engineering & Transportation
Health, Fitness & Dieting
History
Humor & Entertainment
Law
LGBTQ+ Books
Literature & Fiction
Medical Books
Mystery, Thriller & Suspense
Parenting & Relationships
Politics & Social Sciences
Reference
Religion & Spirituality
Romance
Science & Math
Science Fiction & Fantasy
Self-Help
Sports & Outdoors
Teen & Young Adult
Test Preparation

Travel