

Introduction To Computing Using Python Perkovic

Presents standard numerical approaches for solving common mathematical problems in engineering using Python. Covers the most common numerical calculations used by engineering students Covers Numerical Differentiation and Integration, Initial Value Problems, Boundary Value Problems, and Partial Differential Equations Focuses on open ended, real world problems that require students to write a short report/memo as part of the solution process Includes an electronic download of the Python codes presented in the book

Since the invention of computers or machines, their capability to perform various tasks has experienced an exponential growth. Humans have developed the power of computer systems in terms of their diverse working domains, their increasing speed, and reducing size with respect to time.

Perkovic's Introduction to Programming Using Python is more than just an introduction to programming. It is an inclusive introduction to Computer Science that takes the pedagogical approach of "the right tool for the job at the right moment," and focuses on application development. The approach is hands-on and problem-oriented, with practice problems and solutions appearing throughout the text. The text is imperative-first, but does not shy away from discussing objects early where appropriate. Discussions of user-defined classes and Object-Oriented Programming appear later in the text, when students have more background and concepts can be motivated. Chapters include an introduction to problem solving techniques and classical algorithms, problem-solving and programming and ways to apply core skills to application development. This edition also includes examples and practice problems provided within a greater variety of domains. An additional chapter of Case Studies is exclusive to the Wiley E-Text, providing students with real life applications using the concepts and tools covered in the chapters.

Introduction to Computer Science Using Python: A Computational Problem-Solving Focus, recommended by Guido van Rossum, the creator of Python ("This is not your average Python book...I think this book is a great text for anyone teaching CS1"). With a focus on computational problem solving from Chapter 1, this text provides numerous hands-on exercises and examples, each chapter ending with a significant-size program demonstrating the step-by-step process of program development, testing, and debugging. A final chapter includes the history of computing, starting with Charles Babbage, containing over 65 historical images. An end-of-book Python 3 Programmers' Reference is also included for quick lookup of Python details. Extensive instructor materials are provided for those adopting for classroom use, including an instructors' manual, over 1,000 well-developed slides covering all fundamental topics of each chapter, source code, and test bank.

Computer science studies how to describe, predict properties of, and efficiently implement information processes. This book introduces the most important ideas in computing using the Scheme and Python programming languages. It focuses on how to describe information processes by defining procedures, how to analyze the costs required to carry out a procedure, and the fundamental limits of what can and cannot be computed mechanically.

Integrating Python with Leading Computer Forensic Platforms takes a definitive look at how and why the integration of Python advances the field of digital forensics. In addition, the book includes practical, never seen Python examples that can be immediately put to use. Noted author Chet Hosmer demonstrates how to extend four key Forensic Platforms using Python, including EnCase by Guidance Software, MPE+ by AccessData, The Open Source Autopsy/SleuthKit by Brian Carrier and WetStone Technologies, and Live Acquisition and Triage Tool US-LATT. This book is for practitioners, forensic investigators, educators, students, private investigators, or anyone advancing digital forensics for investigating cybercrime. Additionally, the open source availability of the examples allows for sharing and growth within the industry. This book is the first to provide details on how to directly integrate Python into key forensic platforms. Provides hands-on tools, code samples, detailed instruction, and documentation that can be immediately put to use Shows how to integrate Python with popular digital forensic platforms, including EnCase, MPE+, The Open Source Autopsy/SleuthKit, and US-LATT Presents complete coverage of how to use Open Source Python scripts to extend and modify popular digital forensic Platforms

This book is suitable for use in a university-level first course in computing (CS1), as well as the increasingly popular course known as CS0. It is difficult for many students to master basic concepts in computer science and programming. A large portion of the confusion can be blamed on the complexity of the tools and materials that are traditionally used to teach CS1 and CS2. This textbook was written with a single overarching goal: to present the core concepts of computer science as simply as possible without being simplistic.

Students with little or no prior knowledge of programming: students of secondary schools, university students, students of vocational schools, or simply anyone interested in learning programming. The only preliminary requirement is the ability to use a personal computer and very basic knowledge in mathematics.

The text is imperative-first, but does not shy away from discussing objects early where appropriate. Discussions of user-defined classes and Object-Oriented Programming appear later in the text, when students have more background and concepts can be motivated. Chapters include an introduction to problem solving techniques and classical algorithms, problem-solving and programming and ways to apply core skills to application development. This edition also includes examples and practice problems provided within a greater variety of domains.

NOTE: You are purchasing a standalone product; MyProgrammingLab does not come packaged with this content. If you would like to purchase both the physical text and MyProgrammingLab search for 0134059840 / 9780134059846 Introduction to Computing and Programming in Python plus MyProgrammingLab with Pearson eText -- Access Card Package, 4/e Package consists of: 0205891454 / 9780205891450 MyProgrammingLab with Pearson eText -- Access Card -- for Introduction to Computing and Programming in Python 0134025547 / 9780134025544

Introduction to Computing and Programming in Python, 4/e MyProgrammingLab should only be purchased when required by an instructor. Social Computing and Programming with Python Introduction to Computing and Programming in Python is a uniquely researched and up-to-date volume that is widely recognized for its successful introduction to the subject of Media Computation. Emphasizing creativity, classroom interaction, and in-class programming examples, Introduction to Computing and Programming in Python takes a bold and unique approach to computation that engages students and applies the subject matter to the relevancy of digital media. The Fourth Edition teaches students to program in an effort to communicate via social computing outlets, providing a unique approach that serves the interests of a broad range of students. Also Available with MyProgrammingLab® This title is also available with MyProgrammingLab -- an online homework, tutorial, and assessment program designed to work with this text to engage students and improve results. Within its structured environment, students practice what they learn, test their understanding, and pursue a personalized study plan that helps them better absorb course material and understand difficult concepts. Students, if interested in purchasing this title with MyProgrammingLab, ask your instructor for the correct package ISBN and Course ID. Instructors, contact your Pearson representative for more information.

Image Processing and Acquisition using Python provides readers with a sound foundation in both image acquisition and image processing—one of the first books to integrate these topics together. By improving readers' knowledge of image acquisition techniques and corresponding image processing, the book will help them perform experiments more effectively and cost efficiently as well as analyze and measure more accurately. Long recognized as one of the easiest languages for non-programmers to learn, Python is used in a variety of practical examples. A refresher for more experienced readers, the first part of the book presents an introduction to Python, Python modules, reading and writing images using Python, and an introduction to images. The second part discusses the basics of image processing, including pre/post processing using filters, segmentation, morphological operations, and measurements. The second part describes image acquisition using various modalities, such as x-ray, CT, MRI, light microscopy, and electron microscopy. These modalities encompass most of the common image acquisition methods currently used by researchers in academia and industry. Features Covers both the physical methods of obtaining images and the analytical processing methods required to understand the science behind the images. Contains many examples, detailed derivations, and working Python examples of the techniques. Offers practical tips on image acquisition and processing. Includes numerous exercises to test the reader's skills in Python programming and image processing, with solutions to selected problems, example programs, and images available on the book's web page. New to this edition Machine learning has become an indispensable part of image processing and computer vision, so in this new edition two new chapters are included: one on neural networks and the other on convolutional neural networks. A new chapter on affine transform and many new algorithms. Updated Python code aligned to the latest version of modules.

It is an inclusive introduction to Computer Science that takes the pedagogical approach of "the right tool for the job at the right moment," and focuses on application development. The approach is hands-on and problem-oriented, with practice problems and solutions appearing throughout the text. The text is imperative-first, but does not shy away from discussing objects early where appropriate. Discussions of user-defined classes and Object-Oriented Programming appear later in the text, when students have more background and concepts can be motivated. Chapters include an introduction to problem solving techniques and classical algorithms, problem-solving and programming and ways to apply core skills to application development.

Welcome to computer science in the 21st century. Did you ever wonder how computers represent DNA? How they can download a web page containing population data and analyze it to spot trends? Or how they can change the colors in a color photograph? If so, this book is for you. By the time you're done, you'll know how to do all of that and a lot more. And Python makes it easy and fun. Computers are used in every part of science from ecology to particle physics. This introduction to computer science continually reinforces those ties by using real-world science problems as examples. Anyone who has taken a high school science class will be able to follow along as the book introduces the basics of programming, then goes on to show readers how to work with databases, download data from the web automatically, build graphical interfaces, and most importantly, how to think like a professional programmer. Topics covered include: Basic elements of programming from arithmetic to loops and if statements. Using functions and modules to organize programs. Using lists, sets, and dictionaries to organize data. Designing algorithms systematically. Debugging things when they go wrong. Creating and querying databases. Building graphical interfaces to make programs easier to use. Object-oriented programming and programming patterns.

NOTE: You are purchasing a standalone product; MyProgrammingLab does not come packaged with this content. If you would like to purchase both the physical text and MyProgrammingLab search for ISBN-10: 0133050556/ISBN-13: 9780133050554. That package includes ISBN-10: 0132747189/ISBN-13: 9780132747189 and ISBN-10: 0133019861/ISBN-13: 9780133019865 . MyProgrammingLab should only be purchased when required by an instructor. Introduction to Programming Using Python is intended for use in the introduction to programming course. Daniel Liang is known for his "fundamentals-first" approach to teaching programming concepts and techniques. "Fundamentals-first" means that students learn fundamental programming concepts like selection statements, loops, and functions, before moving into defining classes. Students learn basic logic and programming concepts before moving into object-oriented programming, and GUI programming. Another aspect of Introduction to Programming Using Python is that in addition to the typical programming examples that feature games and some math, Liang gives an example or two early in the chapter that uses a simple graphic to engage the students. Rather than asking them to average 10 numbers together, they learn the concepts in the context of a fun example that generates something visually interesting. Using the graphics examples is optional in this textbook. Turtle graphics can be used in Chapters 1-5 to introduce the fundamentals of programming and Tkinter can be used for developing comprehensive graphical user interfaces and for learning object-oriented programming.

Perkovic's Introduction to Computing Using Python: An Application Development Focus, 2nd Edition is more than just an introduction to programming. It is an inclusive introduction to Computer Science that takes the pedagogical approach of "the right tool for the job at the right moment," and focuses on application development. The approach is hands-on and problem-oriented, with practice problems and solutions appearing throughout the text. The text is imperative-first, but does not shy away from discussing objects early where appropriate. Discussions of user-defined classes and Object-Oriented Programming appear later in the text, when students have more background and concepts can be motivated. Chapters include an introduction to problem solving techniques and classical algorithms, problem-solving and programming and ways to apply core skills to application development. This edition also includes examples and practice problems provided within a greater variety of domains. It also includes case studies integrated into additional chapters, providing students with real life applications using the concepts and tools covered in the chapters.

The new edition of an introduction to the art of computational problem solving using Python. This book introduces students with little or no prior programming experience to the art of computational problem solving using Python and various Python libraries, including numpy, matplotlib, random, pandas, and sklearn. It provides students with skills that will enable them to make productive use of computational techniques, including some of the tools and techniques of data science for using computation to model and interpret data as well as substantial material on machine learning. All of the code in the book and an errata sheet are available on the book's web page on the MIT Press website.

For courses in Computer Programming with Python. Social Computing and Programming with Python Introduction to Computing and Programming in Python is a uniquely researched and up-to-date volume that is widely recognized for its successful introduction to the subject of Media Computation. Emphasizing creativity, classroom interaction, and in-class programming examples, Introduction to Computing and Programming in Python takes a bold and unique approach to computation that engages students and applies the subject matter to the relevancy of digital media. The Fourth Edition teaches students to program in an effort to communicate via social computing outlets, providing a unique approach that serves the interests of a broad range of students. MyProgrammingLab® not included. Students, if MyProgrammingLab is a recommended/mandatory component of the course, please ask your instructor for the correct ISBN and course ID. MyProgrammingLab should only be purchased when required by an instructor. Instructors, contact your Pearson representative for more information. MyProgrammingLab is an online homework, tutorial, and assessment product designed to personalize learning and improve results. With a wide range of interactive, engaging, and assignable activities, students are encouraged to actively learn and retain tough course concepts.

Introduction to Computing Using Python is more than just an introduction to programming. It is an inclusive introduction to Computer Science that takes the pedagogical approach of "the right tool for the job at the right moment," and focuses on application development. The approach is hands-on and problem-oriented, with practice problems and solutions appearing throughout the text. The text is imperative-first, but does not shy away from discussing objects early where appropriate. Discussions of user-defined classes and Object-Oriented Programming appear later in the text, when students have more background and concepts can be motivated. Chapters include an introduction to problem solving techniques and classical algorithms, problem-solving and programming and ways to apply core skills to application development. This edition also includes examples and practice problems provided within a greater variety of domains. An additional chapter of Case Studies is exclusive to the Wiley E-Text, providing students with real life applications using the concepts and tools covered in the chapters.

This book is an introduction to the computational methods used in physics and other related scientific fields. It is addressed to an audience that has already been exposed to the introductory level of college physics, usually taught during the first two years of an undergraduate program in science and engineering. It assumes no prior knowledge of numerical analysis, programming or computers and teaches whatever is necessary for the solution of the problems addressed in the text. C++ is used for programming the core programs and data analysis is performed using the powerful tools of the GNU/Linux environment. All the necessary software is open source and freely available. The book starts with very simple problems in particle motion and ends with an in-depth discussion of advanced techniques used in Monte Carlo simulations in statistical mechanics. The level of instruction rises slowly, while discussing problems like the diffusion equation, electrostatics on the plane, quantum mechanics and random walks.

Computational Nuclear Engineering and Radiological Science Using Python provides the necessary knowledge users need to embed more modern computing techniques into current practices, while also helping practitioners replace Fortran-based implementations with higher level languages. The book is especially unique in the market with its implementation of Python into nuclear engineering methods, seeking to do so by first teaching the basics of Python, then going through different techniques to solve systems of equations, and finally applying that knowledge to solve problems specific to nuclear engineering. Along with examples of code and end-of-chapter problems, the book is an asset to novice programmers in nuclear engineering and radiological sciences, teaching them how to analyze complex systems using modern computational techniques. For decades, the paradigm in engineering education, in particular, nuclear engineering, has been to teach Fortran along with numerical methods for solving engineering problems. This has been slowly changing as new codes have been written utilizing modern languages, such as Python, thus resulting in a greater need for the development of more modern computational skills and techniques in nuclear engineering. Offers numerical methods as a tool to solve specific problems in nuclear engineering Provides examples on how to simulate different problems and produce graphs using Python Supplies accompanying codes and data on a companion website, along with solutions to end-of-chapter problems Harness the power of multiple computers using Python through this fast-paced informative guide About This Book- You'll learn to write data processing programs in Python that are highly available, reliable, and fault tolerant- Make use of Amazon Web Services along with Python to establish a powerful remote computation system- Train Python to handle data-intensive and resource hungry applications Who This Book Is For This book is for Python developers who have developed Python programs for data processing and now want to learn how to write fast, efficient programs that perform CPU-intensive data processing tasks. What You Will Learn- Get an introduction to parallel and distributed computing- See synchronous and asynchronous programming- Explore parallelism in Python- Distributed application with Celery- Python in the Cloud- Python on an HPC cluster- Test and debug distributed applications In Detail CPU-intensive data processing tasks have become crucial considering the complexity of the various big data applications that are used today. Reducing the CPU utilization per process is very important to improve the overall speed of applications. This book will teach you how to perform parallel execution of computations by distributing them across multiple processors in a single machine, thus improving the overall performance of a big data processing task. We will cover synchronous and asynchronous models, shared memory and file systems, communication between various processes, synchronization, and more. Style and Approach This example based, step-by-step guide will show you how to make the best of your hardware configuration using Python for distributing applications.

Serieuze wetenschappelijke antwoorden op belangrijke hypothetische vragen Wat als? geeft hilarische en informatieve antwoorden op belangrijke vragen waar je nooit aan zou denken. Vragen als: • Als iedereen op aarde een paar weken bij elkaar uit de buurt blijft, is de verkoudheid dan niet de wereld uit? / br• Wat als je een baseball probeert te raken die met 90% van de snelheid van het licht op je af komt? / br• Van welke hoogte moet je een steak laten vallen om hem gaar te laten zijn als hij de grond raakt? / br• Als mijn printer letterlijk geld kan drukken, heeft dat dan grote gevolgen voor de wereld? /

br• Wat gebeurt er als iedereen op aarde zo dicht mogelijk bij elkaar gaat staan en opspringt, waarna iedereen op hetzelfde moment neerkomt? De antwoorden van Munroe zijn kleine meesterwerken van duidelijkheid en hilariteit aangevuld met zijn kenmerkende tekeningen. De antwoorden voorspellen vaak volledige vernietiging van de mensheid of op z'n minst een heel grote explosie. Randall Munroe is de bedenker van xkcd.com, de razend populaire website die elke week door miljoenen mensen wordt bezocht. Na z'n studie fysica bouwde Munroe robots bij NASA, waarna hij in 2006 fulltime striptekenaar werd.

Cowan's Introduction to Programming Using Python, is more than just an introduction to programming. It is an inclusive introduction to Computer Science that takes the pedagogical approach of "the right tool for the job at the right moment," and focuses on application development. The approach is hands-on and problem-oriented, with practice problems and solutions appearing throughout the text. The text is imperative-first, but does not shy away from discussing objects early where appropriate. Discussions of user-defined classes and Object-Oriented Programming appear later in the text, when students have more background and concepts can be motivated. Chapters include an introduction to problem solving techniques and classical algorithms, problem-solving and programming and ways to apply core skills to application development.

This book 'Introduction to Computing and Problem Solving with Python' will help every student, teacher and researcher to understand the computing basics and advanced Python Programming language. The Python programming topics include the reserved keywords, identifiers, variables, operators, data types and their operations, flow control techniques which include decision making and looping, modules, files and exception handling techniques. Advanced topics like Python regular expressions, Database Programming and Object Oriented Programming concepts are also covered in detail. All chapters have worked out programs, illustrations, review and frequently asked interview questions. The simple style of presentation makes this a friend for self-learners. More than 300 solved lab exercises available in this book is tested in Python 3.4.3 version for Windows. The book covers syllabus for more than 35 International Universities and 45 Indian universities like Dr. APJ Abdul Kalam Technological University, Christ University, Savitribai Phule Pune University, University of Delhi, University of Calicut, Mahatma Gandhi University, University of Mumbai, AICTE, CBSE, MIT, University of Virginia, University of Chicago, University of Toronto, Technical University of Denmark etc.

Experimental Animation: From Analogue to Digital, focuses on both experimental animation's deep roots in the twentieth century, and its current position in the twenty-first century media landscape. Each chapter incorporates a variety of theoretical lenses, including historical, materialist, phenomenological and scientific perspectives. Acknowledging that process is a fundamental operation underlining experimental practice, the book includes not only chapters by international academics, but also interviews with well-known experimental animation practitioners such as William Kentridge, Jodie Mack, Larry Cuba, Martha Colburn and Max Hattler. These interviews document both their creative process and thoughts about experimental animation's ontology to give readers insight into contemporary practice. Global in its scope, the book features and discusses lesser known practitioners and unique case studies, offering both undergraduate and graduate students a collection of valuable contributions to film and animation studies.

The new edition of an introductory text that teaches students the art of computational problem solving, covering topics ranging from simple algorithms to information visualization. This book introduces students with little or no prior programming experience to the art of computational problem solving using Python and various Python libraries, including PyLab. It provides students with skills that will enable them to make productive use of computational techniques, including some of the tools and techniques of data science for using computation to model and interpret data. The book is based on an MIT course (which became the most popular course offered through MIT's OpenCourseWare) and was developed for use not only in a conventional classroom but in a massive open online course (MOOC). This new edition has been updated for Python 3, reorganized to make it easier to use for courses that cover only a subset of the material, and offers additional material including five new chapters. Students are introduced to Python and the basics of programming in the context of such computational concepts and techniques as exhaustive enumeration, bisection search, and efficient approximation algorithms. Although it covers such traditional topics as computational complexity and simple algorithms, the book focuses on a wide range of topics not found in most introductory texts, including information visualization, simulations to model randomness, computational techniques to understand data, and statistical techniques that inform (and misinform) as well as two related but relatively advanced topics: optimization problems and dynamic programming. This edition offers expanded material on statistics and machine learning and new chapters on Frequentist and Bayesian statistics.

Cognitive Computing is a new topic which aims to simulate human thought processes using computers that self-learn through data mining, pattern recognition, and natural language processing. This book focuses on the applications of Cognitive Computing in areas like Robotics, Blockchain, Deep Learning, and Wireless Technologies. This book covers the basics of Green Computing, discusses Cognitive Science methodologies in Robotics, Computer Science, Wireless Networks, and Deep Learning. It goes on to present empirical data and research techniques, modelling techniques and offers a data-driven approach to decision making and problem solving. This book is written for researchers, academicians, undergraduate and graduate students, and industry persons who are working on current applications of Cognitive Computing.

This book addresses students and young researchers who want to learn to use numerical modeling to solve problems in geodynamics. Intended as an easy-to-use and self-learning guide, readers only need a basic background in calculus to approach most of the material. The book difficulty increases very gradually, through four distinct parts. The first is an introduction to the Python techniques necessary to visualize and run vectorial calculations. The second is an overview with several examples on classical Mechanics with examples taken from standard introductory physics books. The third part is a detailed description of how to write Lagrangian, Eulerian and Particles in Cell codes for solving linear and non-linear continuum mechanics problems. Finally the last one address advanced techniques like tree-codes, Boundary Elements, and illustrates several applications to Geodynamics. The entire book is organized around numerous examples in Python, aiming at encouraging the reader to learn by experimenting and experiencing, not by theory.

Guzdial introduces programming as a way of creating and manipulating media in a context familiar and intriguing to today's readers. Starts readers with actual programming early on. Puts programming in a relevant context (Computing for Communications). Includes implementing Photoshop-like effects, reversing/splicing sounds, creating animations. Acknowledges that readers in this audience care about the Web; introduces HTML and covers writing programs that generate HTML. Uses the Web as a Data Source; shows readers how to read from files, but also how to write programs to directly read Web pages and distill information from there for use in other calculations, other Web pages, etc. (examples include temperature from a weather page, stock prices from a financials page). A comprehensive guide for anyone interested in learning the basics of programming with one of the best web languages, Python.

Enhance your data analysis and predictive modeling skills using popular Python tools Key Features Cover all fundamental libraries for operation and manipulation of Python for data analysis Implement real-world datasets to perform predictive analytics with Python Access modern data analysis techniques and detailed code with scikit-learn and SciPy Book Description Python is one of the most common and popular languages preferred by leading data analysts and statisticians for working with massive datasets and complex data visualizations. Become a Python Data Analyst introduces Python's most essential

tools and libraries necessary to work with the data analysis process, right from preparing data to performing simple statistical analyses and creating meaningful data visualizations. In this book, we will cover Python libraries such as NumPy, pandas, matplotlib, seaborn, SciPy, and scikit-learn, and apply them in practical data analysis and statistics examples. As you make your way through the chapters, you will learn to efficiently use the Jupyter Notebook to operate and manipulate data using NumPy and the pandas library. In the concluding chapters, you will gain experience in building simple predictive models and carrying out statistical computation and analysis using rich Python tools and proven data analysis techniques. By the end of this book, you will have hands-on experience performing data analysis with Python. What you will learn Explore important Python libraries and learn to install Anaconda distribution Understand the basics of NumPy Produce informative and useful visualizations for analyzing data Perform common statistical calculations Build predictive models and understand the principles of predictive analytics Who this book is for Become a Python Data Analyst is for entry-level data analysts, data engineers, and BI professionals who want to make complete use of Python tools for performing efficient data analysis. Prior knowledge of Python programming is necessary to understand the concepts covered in this book

Leer om niet langer je eigen vijand te zijn en iedere minuut voluit te leven! Veel mensen kennen het gevoel tekort te schieten maar al te goed. Er is niet veel voor nodig horen hoe goed iemand anders is, zelf bekritiseerd worden, een woordenwisseling, een fout op het werk om te denken dat we niet goed genoeg zijn. Dit kan leiden tot zelfveroordeling, relatieproblemen, perfectionisme, eenzaamheid en overwerk. Zelfaanvaarding is een proces dat een leven lang duurt. In Het leven liefhebben door acceptatie beschrijft Tara Brach op een open en eerlijke manier hoe zij haar weg heeft gevonden. Via haar persoonlijke verhaal en dat van haar cliënten en leerlingen, geleide meditatie, gedichten en citaten weet zij tot de kern van het probleem door te dringen. Want pas als je begrijpt hoe het gevoel van onwaardigheid is ontstaan, kun je verbinding maken met je echte ik en jezelf leren vertrouwen en omarmen. Tara Brach is psychotherapeut en toonaangevend leraar op het gebied van mindfulness, emotionele genezing en spiritueel ontwaken. Zij is oprichter van de Insight Meditation Community en geeft zeer drukbezochte workshops in binnen- en buitenland. www.tarabrach.com

Introduction to Computing and Programming in Python, 3e, uses multimedia applications to motivate introductory computer science majors or non-majors. The book's hands-on approach shows how programs can be used to build multimedia computer science applications that include sound, graphics, music, pictures, and movies. The students learn a key set of computer science tools and topics, as well as programming skills; such as how to design and use algorithms, and practical software engineering methods. The book also includes optional coverage of HCI, as well as rudimentary data structures and databases using the user-friendly Python language for implementation. Authors Guzdial and Ericson also demonstrate how to communicate compatibly through networks and do concurrent programming. 0133591522 / 9780133591521 Introduction to Computing and Programming in Python & MyProgrammingLab with eText Package Package consists of 0132923513 / 9780132923514 Introduction to Computing and Programming in Python 0133590747 / 9780133590746 MyProgrammingLab with eText -- Access Code Card -- for Introduction to Computing and Programming in Python

Learn to use IPython and Jupyter Notebook for your data analysis and visualization work. Key Features Leverage the Jupyter Notebook for interactive data science and visualization Become an expert in high-performance computing and visualization for data analysis and scientific modeling A comprehensive coverage of scientific computing through many hands-on, example-driven recipes with detailed, step-by-step explanations Book Description Python is one of the leading open source platforms for data science and numerical computing. IPython and the associated Jupyter Notebook offer efficient interfaces to Python for data analysis and interactive visualization, and they constitute an ideal gateway to the platform. IPython Interactive Computing and Visualization Cookbook, Second Edition contains many ready-to-use, focused recipes for high-performance scientific computing and data analysis, from the latest IPython/Jupyter features to the most advanced tricks, to help you write better and faster code. You will apply these state-of-the-art methods to various real-world examples, illustrating topics in applied mathematics, scientific modeling, and machine learning. The first part of the book covers programming techniques: code quality and reproducibility, code optimization, high-performance computing through just-in-time compilation, parallel computing, and graphics card programming. The second part tackles data science, statistics, machine learning, signal and image processing, dynamical systems, and pure and applied mathematics. What you will learn Master all features of the Jupyter Notebook Code better: write high-quality, readable, and well-tested programs; profile and optimize your code; and conduct reproducible interactive computing experiments Visualize data and create interactive plots in the Jupyter Notebook Write blazingly fast Python programs with NumPy, ctypes, Numba, Cython, OpenMP, GPU programming (CUDA), parallel IPython, Dask, and more Analyze data with Bayesian or frequentist statistics (Pandas, PyMC, and R), and learn from actual data through machine learning (scikit-learn) Gain valuable insights into signals, images, and sounds with SciPy, scikit-image, and OpenCV Simulate deterministic and stochastic dynamical systems in Python Familiarize yourself with math in Python using SymPy and Sage: algebra, analysis, logic, graphs, geometry, and probability theory Who this book is for This book is intended for anyone interested in numerical computing and data science: students, researchers, teachers, engineers, analysts, and hobbyists. A basic knowledge of Python/NumPy is recommended. Some skills in mathematics will help you understand the theory behind the computational methods.

Previous edition: published as by Jennifer Campbell ... [et al]. 2009.

In Materiaalkunde komen alle belangrijke materialen die toegepast worden in werktuigbouwkundige constructies aan de orde, zoals metalen, kunststoffen en keramiek. Per materiaalgroep behandelen de auteurs: · de belangrijkste eigenschappen; · de manier van verwerking; · de beperkingen; · de belangrijkste keuzeaspecten met betrekking tot constructies; · de manier van specificatie in een technische tekening of een ontwerp. De eerste editie van Materiaalkunde verscheen alweer dertig jaar geleden. In de tussentijd is het voortdurend aangepast aan de nieuwste ontwikkelingen en het mag dan ook met recht een klassieker genoemd worden.

[Copyright: 28f71989fd6652b0e962538d08bf40b0](https://www.amazon.com/dp/0133591522)