Machine Language Programming Cookbook

Over 50 hands-on recipes that will help you develop amazing real-time applications using GPIO, RS232, ADC, DAC, timers, audio codecs, graphics LCD, and a touch screen About This Book This book focuses on programming embedded systems using a practical approach Examples show how to use bitmapped graphics and manipulate digital audio to produce amazing games and other multimedia applications The recipes in this book are written using ARM's MDK Microcontroller Development Kit which is the most comprehensive and accessible development solution Who This Book Is For This book is aimed at those with an interest in designing and programming embedded systems. These could include electrical engineers or computer programmers who want to get started with microcontroller applications using the ARM Cortex-M4 architecture in a short time frame. The book's recipes can also be used to support students learning embedded programming for the first time. Basic knowledge of programming using a high level language is essential but those familiar with other high level languages such as Python or Java should not have too much difficulty picking up the basics of embedded C programming. What You Will Learn Use ARM's uVision MDK to configure the microcontroller run time environment (RTE), create projects and compile download and run simple programs on an evaluation board. Use and extend device family packs to configure I/O peripherals. Develop multimedia applications using the touchscreen and audio codec beep generator. Configure the codec to stream digital audio and design digital filters to create amazing audio effects. Write multi-threaded programs using ARM's real time operating system (RTOS). Write critical sections of code in assembly language and integrate these with functions written in C. Fix problems using ARM's debugging tool to set breakpoints and examine variables. Port uVision projects to other open source development environments. In Detail Embedded microcontrollers are at the core of many everyday electronic devices. Electronic automotive systems rely on these devices for engine management, anti-lock brakes, in car entertainment, automatic transmission, active suspension, satellite navigation, etc. The so-called internet of things drives the market for such technology, so much so that embedded cores now represent 90% of all processor's sold. The ARM Cortex-M4 is one of the most powerful microcontrollers on the market and includes a floating point unit (FPU) which enables it to address applications. The ARM Cortex-M4 Microcontroller Cookbook provides a practical introduction to programming an embedded microcontroller architecture. This book attempts to address this through a series of recipes that develop embedded applications targeting the ARM-Cortex M4 device family. The recipes in this book have all been tested using the Keil MCBSTM32F400 board. This board includes a small graphic LCD touchscreen (320x240 pixels) that can be used to create a variety of 2D gaming applications. These motivate a younger audience and are used throughout the book to illustrate particular hardware peripherals and software concepts. C language is used predominantly throughout but one chapter is devoted to recipes involving assembly language. Programs are mostly written using ARM's free microcontroller development kit (MDK) but for those looking for open source development environments the book also shows how to configure the ARM-GNU toolchain. Some of the recipes described in the book are the basis for laboratories and assignments undertaken by undergraduates. Style and approach The ARM Cortex-M4 Cookbook is a practical guide full of hands-on recipes. It follows a step-by-step approach that allows you to find, utilize and learn ARM concepts quickly.

Quick fixes to your common C# programming problems, with a focus on C# 6.0 About This Book Unique recipes for C#, that places it in its real-world context. You will be able to get yourself out of any coding-corner youve backed yourself into. All code samples available through GitHub to bring C#. In line with modern development workflows, written to the latest specification of C# 6.0. Who This Book Is For The book is aimed at developers who have basic familiarity with C# programming and will know the VS 2015 environment. What You Will Learn Write better and less code to achieve the same result as in previous versions of C#. Generate tests from the Code Contracts for mission critical methods. Integrate code in Visual Studio with GitHub. Create a web application in Azure. Design and build a microservice architecture Demystify the Rx library using Reactive extensions Write high performing codes in C# and understanding multi-threading. Security and Debugging. Implement Code Contracts on code in Visual Studio. In Detail During your application development workflow, there is always a moment when you need to get out of a tight spot. Through a recipe-based approach, this book will help you overcome common programming problems and get your applications ready to face the modern world. We start with C# 6, giving you hands-on experience with the new language features. Next, we work through the tasks that you perform on a daily basis such as working with strings, generics, and lots more. Gradually, we move on to more advanced topics such as the concept of object-oriented programming, asynchronous programming, reactive extensions, and code contracts. You will learn responsive high performance programming in C# and how to create applications with Azure. Next, we will review the choices available when choosing a source control solution. At the end of the book, we will show you how to create secure and robust code, and will help you ramp up your skills when using the new version of C# 6 and Visual Studio Style and Approach Unique recipe-based guide that will help you gain a solid understanding of the new concepts in C# 6 and Visual Studio Enterprise 2015 in a concise and technically correct manner.

Don Lancaster's Micro Cookbook: Machine language programmingOpenCV 2 Computer Vision Application Programming CookbookOver 50 Recipes to Master this Library of Programming Functions for Real-time Computer VisionPackt Pub Limited

A pragmatic recipe book for acquiring a comprehensive understanding of the complexities and core fundamentals of C++ programming Key FeaturesExplore the latest language and library features of C++20 such as modules, coroutines, concepts, and rangesShed new light on the core concepts in C++ programming, including functions, algorithms, threading, and concurrency, through practical self-contained recipesLeverage C++ features like smart pointers, move semantics, constexpr, and more for increased robustness and performanceBook Description C++ has come a long way to be one of the most widely used general-purpose languages that is fast, efficient, and high-performance at its core. The updated second edition of Modern C++ Programming Cookbook addresses the latest features of C++20, such as modules, concepts, coroutines, and the many additions to the standard library, including ranges and text formatting. The book is organized in the form of practical recipes covering a wide range of problems faced by modern developers. The book also delves into the details of all the core concepts in modern C++ programming, such as functions and classes, iterators and algorithms, streams and the file system, threading and concurrency, smart pointers and move semantics, and many others. It goes into the performance aspects of programming in depth, teaching developers how to write fast and lean code with the help of best practices. Furthermore, the book explores useful patterns and delves into the implementation of many idioms, including pimpl, named parameter, and attorney-client, teaching techniques such as avoiding repetition with the factory pattern. There is also a chapter dedicated to unit testing, where you are introduced to three of the most widely used libraries for C++: Boost.Test, Google Test, and Catch2. By the end of the book, you will be able to effectively leverage the features and techniques of C++11/14/17/20 programming to enhance the performance, scalability, and efficiency of your applications. What you will learnUnderstand the new C++20 language and library features and the problems they solveBecome skilled at using the standard support for threading and concurrency for daily tasksLeverage the standard library and work with containers, algorithms, and iteratorsSolve text searching and replacement problems using regular expressionsWork with different types of strings and learn the various aspects of compilationTake advantage of the file system library to work with files and directoriesImplement various useful patterns and idiomsExplore the widely used testing frameworks for C++Who this book is for The book is designed for entry- or medium-level C++ programmers who have a basic knowledge of C++ and want to master the language and become prolific modern C++ developers. Experienced C++ programmers can leverage this book to strengthen their command of C++ and find a good reference to many language and library features of C++11/14/17/20.

Over 85 recipes to build modular, readable, and testable Golang applications across various domains, 2nd Edition

LabVIEW Graphical Programming

Python Network Programming Cookbook

Virtual Machines

Ruby Cookbook

Don Lancaster's Micro Cookbook: Machine language programming

LabVIEW is an award-winning programming language that allows engineers to create "virtual" instruments on their desktop. This new edition details the powerful features of LabVIEW 8.0. Written in a highly accessible and readable style, LabVIEW Graphical Programming illustrates basic LabVIEW programming techniques, building up to advanced programming concepts. New to this edition is study material for the CLAD and CLD exams.

In this book, you will learn how to use NumPy, Pandas, OpenCV, Scikit-Learn and other libraries to how to plot graph and to process digital image. Then, you will learn how to classify features using Perceptron, Adaline, Logistic Regression (LR), Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF), and K-Nearest Neighbor (KNN) models. You will also learn how to extract features using Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Kernel Principal Component Analysis (KPCA) algorithms and use them in machine learning. In Chapter 1, you will learn: Tutorial Steps To Create A Simple GUI Application, Tutorial Steps To Use Radio Button, Tutorial Steps To Group Radio Buttons, Tutorial Steps To Use CheckBox Widget, Tutorial Steps to Use Two CheckBox Groups, Tutorial Steps to Understand Signals and Slots, Tutorial Steps to Convert Data Types, Tutorial Steps to Use Spin Box Widget, Tutorial Steps to Use ScrollBar and Slider, Tutorial Steps to Use List Widget, Tutorial Steps to Select Multiple List Items in One List Widget and Display It in Another List Widget, Tutorial Steps to Insert Item into List Widget, Tutorial Steps to Use Operations on Widget List, Tutorial Steps to Use Combo Box, Tutorial Steps to Use Calendar Widget and Date Edit, and Tutorial Steps to Use Table Widget. In Chapter 2, you will learn: Tutorial Steps To Create A Simple Line Graph, Tutorial Steps To Create A Simple Line Graph in Python GUI, Tutorial Steps To Create A Simple Line Graph in Python GUI: Part 2, Tutorial Steps To Create Two or More Graphs in the Same Axis, Tutorial Steps To Create Two Axes in One Canvas, Tutorial Steps To Use Two Widgets, Tutorial Steps To Use Two Widgets, Each of Which Has Two Axes, Tutorial Steps To Use Axes With Certain Opacity Levels, Tutorial Steps To Choose Line Color From Combo Box, Tutorial Steps To Calculate Fast Fourier Transform, Tutorial Steps To Create GUI For FFT, Tutorial Steps To Create GUI For FFT With Some Other Input Signals, Tutorial Steps To Create GUI For Noisy Signal, Tutorial Steps To Create GUI For Noisy Signal Filtering, and Tutorial Steps To Create GUI For Wav Signal Filtering. In Chapter 3, you will learn: Tutorial Steps To Convert RGB Image Into Grayscale, Tutorial Steps To Convert RGB Image Into YUV Image, Tutorial Steps To Convert RGB Image Into HSV Image, Tutorial Steps To Filter Image, Tutorial Steps To Display Image Histogram, Tutorial Steps To Display Filtered Image Histogram, Tutorial Steps To Filter Image With CheckBoxes, Tutorial Steps To Implement Image Thresholding, and Tutorial Steps To Implement Adaptive Image Thresholding. You will also learn: Tutorial Steps To Generate And Display Noisy Image, Tutorial Steps To Implement Edge Detection On Image, Tutorial Steps To Implement Image Segmentation Using Multiple Thresholding and K-Means Algorithm, Tutorial Steps To Implement Image Denoising, Tutorial Steps To Detect Face, Eye, and Mouth Using Haar Cascades, Tutorial Steps To Detect Face Using Haar Cascades with PyQt, Tutorial Steps To Detect Eye, and Mouth Using Haar Cascades with PyQt, Tutorial Steps To Extract Detected Objects, Tutorial Steps To Detect Image Features Using Harris Corner Detection, Tutorial Steps To Detect Image Features Using Shi-Tomasi Corner Detection, Tutorial Steps To Detect Features Using Scale-Invariant Feature Transform (SIFT), and Tutorial Steps To Detect Features Using Features from Accelerated Segment Test (FAST). In Chapter 4, In this tutorial, you will learn how to use Pandas, NumPy and other libraries to perform simple classification using perceptron and Adaline (adaptive linear neuron). The dataset used is Iris dataset directly from the UCI Machine Learning Repository. You will learn: Tutorial Steps To Implement Perceptron, Tutorial Steps To Implement Perceptron with PyQt, Tutorial Steps To Implement Adaline (ADAptive LInear NEuron), and Tutorial Steps To Implement Adaline with PyQt. In Chapter 5, you will learn how to use the scikit-learn machine learning library, which provides a wide variety of machine learning algorithms via a user-friendly Python API and to perform classification using perceptron, Adaline (adaptive linear neuron), and other models. The dataset used is Iris dataset directly from the UCI Machine Learning Repository. You will learn: Tutorial Steps To Implement Perceptron Using Scikit-Learn, Tutorial Steps To Implement Perceptron Using Scikit-Learn with PyQt, Tutorial Steps To Implement Logistic Regression Model, Tutorial Steps To Implement Logistic Regression Model Using Scikit-Learn, Tutorial Steps To Implement Logistic Regression Model Using Scikit-Learn with PyQt, Tutorial Steps To Implement Support Vector Machine (SVM) Using Scikit-Learn, Tutorial Steps To Implement Decision Tree (DT) Using Scikit-Learn, Tutorial Steps To Implement Random Forest (RF) Using Scikit-Learn, and Tutorial Steps To Implement K-Nearest Neighbor (KNN) Using Scikit-Learn. In Chapter 6, you will learn how to use Pandas, NumPy, Scikit-Learn, and other libraries to implement different approaches for reducing the dimensionality of a dataset using different feature selection techniques. You will learn about three fundamental techniques that will help us to summarize the information content of a dataset by transforming it onto a new feature subspace of lower dimensionality than the original one. Data compression is an important topic in machine learning, and it helps us to store and analyze the increasing amounts of data that are produced and collected in the modern age of technology. You will learn the following topics: Principal Component Analysis (PCA) for unsupervised data compression, Linear Discriminant Analysis (LDA) as a supervised dimensionality reduction technique for maximizing class separability, Nonlinear dimensionality reduction via Kernel Principal Component Analysis (KPCA). You will learn: 6.1 Tutorial Steps To Implement Principal Component Analysis (PCA), Tutorial Steps To Implement Principal Component Analysis (PCA) Using Scikit-Learn, Tutorial Steps To Implement Principal Component Analysis (PCA) Using Scikit-Learn with PyQt, Tutorial Steps To Implement Linear Discriminant Analysis (LDA), Tutorial Steps To Implement Linear Discriminant Analysis (LDA) with Scikit-Learn, Tutorial Steps To Implement Linear Discriminant Analysis (LDA) Using Scikit-Learn with PyQt, Tutorial Steps To Implement Kernel Principal Component Analysis (KPCA) Using Scikit-Learn, and Tutorial Steps To Implement Kernel Principal Component Analysis (KPCA) Using Scikit-Learn with PyQt. In Chapter 7, you will learn how to use Keras, Scikit-Learn, Pandas, NumPy and other libraries to perform prediction on handwritten digits using MNIST dataset. You will learn: Tutorial Steps To Load MNIST Dataset, Tutorial Steps To Load MNIST Dataset with PyQt, Tutorial Steps To Implement Perceptron With PCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Perceptron With LDA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Perceptron With KPCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Logistic Regression (LR) Model With PCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Logistic Regression (LR) Model With LDA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Logistic Regression (LR) Model With KPCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement , Tutorial Steps To Implement Support Vector Machine (SVM) Model With LDA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Support Vector Machine (SVM) Model With KPCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Decision Tree (DT) Model With PCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Decision Tree (DT) Model With LDA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Decision Tree (DT) Model With KPCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Random Forest (RF) Model With PCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Random Forest (RF) Model With LDA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Random Forest (RF) Model With KPCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement K-Nearest Neighbor (KNN) Model With PCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement K-Nearest Neighbor (KNN) Model With LDA Feature Extractor on MNIST Dataset Using PyQt, and Tutorial Steps To Implement K-Nearest Neighbor (KNN) Model With KPCA Feature Extractor on MNIST Dataset Using PyQt.

While compilers for high-level programming languages are large complex software systems, they have particular characteristics that differentiate them from other software systems. Their functionality is almost completely well-defined – ideally there exist complete precise descriptions of the source and target languages, while additional descriptions of the interfaces to the operating system, programming system and programming environment, and to other compilers and libraries are often available. The implementation of application systems directly in machine language is both difficult and error-prone, leading to programs that become obsolete as quickly as the computers for which they were developed. With the development of higher-level machine-independent programming languages came the need to offer compilers that were able to translate programs into machine language. Given this basic challenge, the different subtasks of compilation have been the subject of intensive research since the 1950s. This book is not intended to be a cookbook for compilers, instead the authors' presentation reflects the special characteristics of compiler design, especially the existence of precise specifications of the subtasks. They invest effort to understand these precisely and to provide adequate concepts for their systematic treatment. This is the first book in a multivolume set, and here the authors describe what a compiler does, i.e., what correspondence it establishes between a source and a target program. To achieve this the authors specify a suitable virtual machine (abstract machine) and exactly describe the compilation of programs of each source language into the language of the associated virtual machine for an imperative, functional, logic and object-oriented programming language. This book is intended for students of computer science. Knowledge of at least one imperative programming language is assumed, while for the chapters on the translation of functional and logic programming languages it would be helpful to know a modern functional language and Prolog. The book is supported throughout with examples, exercises and program fragments.

PROJECT 1: THE APPLIED DATA SCIENCE WORKSHOP: Prostate Cancer Classification and Recognition Using Machine Learning and Deep Learning with Python GUI Prostate cancer is cancer that occurs in the prostate. The prostate is a small walnut-shaped gland in males that produces the seminal fluid that nourishes and transports sperm. Prostate cancer is one of the most common types of cancer. Many prostate cancers grow slowly and are confined to the prostate gland, where they may not cause serious harm. However, while some types of prostate cancer grow slowly and may need minimal or even no treatment, other types are aggressive and can spread quickly. The dataset used in this project consists of 100 patients which can be used to implement the machine learning and deep learning algorithms. The dataset consists of 100 observations and 10 variables (out of which 8 numeric variables and one categorical variable and is ID) which are as follows: Id, Radius, Texture, Perimeter, Area, Smoothness, Compactness, Diagnosis Result, Symmetry, and Fractal Dimension. The models used in this project are K-Nearest Neighbor, Random Forest, Naive Bayes, Logistic Regression, Decision Tree, Support Vector Machine, Adaboost, LGBM classifier, Gradient Boosting, XGB classifier, MLP classifier, and CNN 1D. Finally, you will develop a GUI using PyQt5 to plot boundary decision, ROC, distribution of features, feature importance, cross validation score, and predicted values versus true values, confusion matrix, learning curve, performance of the model, scalability of the model, training loss, and training accuracy. PROJECT 2: THE APPLIED DATA SCIENCE WORKSHOP: Urinary Biomarkers Based Pancreatic Cancer Classification and Prediction Using Machine Learning with Python GUI Pancreatic cancer is an extremely deadly type of cancer. Once diagnosed, the five-year survival rate is less than 10%. However, if pancreatic cancer is caught early, the odds of surviving are much better. Unfortunately, many cases of pancreatic cancer show no symptoms until the cancer has spread throughout the body. A diagnostic test to identify people with pancreatic cancer could be enormously helpful. In a paper by Silvana Debernardi and colleagues, published this year in the journal PLOS Medicine, a multi-national team of researchers sought to develop an accurate diagnostic test for the most common type of pancreatic cancer, called pancreatic ductal adenocarcinoma or PDAC. They gathered a series of biomarkers from the urine of three groups of patients: Healthy controls, Patients with non-cancerous pancreatic conditions, like chronic pancreatitis, and Patients with pancreatic ductal adenocarcinoma. When possible, these patients were age- and sex-matched. The goal was to develop an accurate way to identify patients with pancreatic cancer. The key features are four urinary biomarkers: creatinine, LYVE1, REG1B, and TFF1. Creatinine is a protein that is often used as an indicator of kidney function. YVLE1 is lymphatic vessel endothelial hyaluronan receptor 1, a protein that may play a role in tumor metastasis. REG1B is a protein that may be associated with pancreas regeneration. TFF1 is trefoil factor 1, which may be related to regeneration and repair of the urinary tract. The models used in this project are K-Nearest Neighbor, Random Forest, Naive Bayes, Logistic Regression, Decision Tree, Support Vector Machine, Adaboost, LGBM classifier, Gradient Boosting, XGB classifier, and MLP classifier. Finally, you will develop a GUI using PyQt5 to plot boundary decision, ROC, distribution of features, feature importance, cross validation score, and predicted values versus true values, confusion matrix, learning curve, performance of the model, scalability of the model, training loss, and training accuracy. PROJECT 3: DATA SCIENCE CRASH COURSE: Voice Based Gender Classification and Prediction Using Machine Learning and Deep Learning with Python GUI This dataset was created to identify a voice as male or female, based upon acoustic properties of the voice and speech. The dataset consists of 3,168 recorded voice samples, collected from male and female speakers. The voice samples are pre-processed by acoustic analysis in R using the seewave and tuneR packages, with an analyzed frequency range of 0hz-280hz (human vocal range). The following acoustic properties of each voice are measured and included within the CSV: meanfreq: mean frequency (in kHz); sd: standard deviation of frequency; median: median frequency (in kHz); Q25: first quantile (in kHz); Q75: third quantile (in kHz); IQR: interquantile range (in kHz); skew: skewness; kurt: kurtosis; sp.ent: spectral entropy; sfm: spectral flatness; mode: mode frequency; centroid: frequency centroid (see specprop); peakf: peak frequency (frequency with highest energy); meanfun: average of fundamental frequency measured across acoustic signal; minfun: minimum fundamental frequency measured across acoustic signal; maxfun: maximum fundamental frequency measured across acoustic signal; meandom: average of dominant frequency measured across acoustic signal; mindom: minimum of dominant frequency measured across acoustic signal; maxdom: maximum of dominant frequency measured across acoustic signal; dfrange: range of dominant frequency measured across acoustic signal; modindx: modulation index. Calculated as the accumulated absolute difference between adjacent measurements of fundamental frequencies divided by the frequency range; and label: male or female. The models used in this project are K-Nearest Neighbor, Random Forest, Naive Bayes, Logistic Regression, Decision Tree, Support Vector Machine, Adaboost, LGBM classifier, Gradient Boosting, XGB classifier, MLP classifier, and CNN 1D. Finally, you will develop a GUI using PyQt5 to plot boundary decision, ROC, distribution of features, feature importance, cross validation score, and predicted values versus true values, confusion matrix, learning curve, performance of the model, scalability of the model, training loss, and training accuracy. PROJECT 4: DATA SCIENCE CRASH COURSE: Thyroid Disease Classification and Prediction Using Machine Learning and Deep Learning with Python GUI Thyroid disease is a general term for a medical condition that keeps your thyroid from making the right amount of hormones. Thyroid typically makes hormones that keep your body functioning normally. When the thyroid makes too much thyroid hormone, body uses energy too quickly. The two main types of thyroid disease are hypothyroidism and hyperthyroidism. Both conditions can be caused by other diseases that impact the way the thyroid gland works. Dataset used in this project was from Garavan Institute Documentation as given by Ross Quinlan 6 databases from the Garavan Institute in Sydney, Australia. Approximately the following for each database: 2800 training (data) instances and 972 test instances. This dataset contains plenty of missing data, while 29 or so attributes, either Boolean or continuously-valued. The models used in this project are K-Nearest Neighbor, Random Forest, Naive Bayes, Logistic Regression, Decision Tree, Support Vector Machine, Adaboost, LGBM classifier, Gradient Boosting, XGB classifier, MLP classifier, and CNN 1D. Finally, you will develop a GUI using PyQt5 to plot boundary decision, ROC, distribution of features, feature importance, cross validation score, and predicted values versus true values, confusion matrix, performance of the model, scalability of the model, training loss, and training accuracy.

Compiler Design
Raspberry Pi 3 Cookbook for Python Programmers
Raspberry Pi Cookbook for Python Programmers
OpenCV 2 Computer Vision Application Programming Cookbook
LEARN FROM SCRATCH MACHINE LEARNING WITH PYTHON GUI
Advanced C++ Programming Cookbook

If you need help writing programs in Python 3, or want to update older Python 2 code, this book is just the ticket. Packed with practical recipes written and tested with Python 3.3, this unique cookbook is for experienced Python programmers who want to focus on modern tools and idioms. Inside, you'll find complete recipes for more than a dozen topics, covering the core Python language as well as tasks common to a wide variety of application domains. Each recipe contains code samples you can use in your projects right away, along with a discussion about how and why the solution works. Topics include: Data Structures and Algorithms Strings and Text Numbers, Dates, and Times Iterators and Generators Files and I/O Data Encoding and Processing Functions Classes and Objects Metaprogramming Modules and Packages Network and Web Programming Concurrency Utility Scripting and System Administration Testing, Debugging, and Exceptions C Extensions

An easy-to-follow guide full of hands-on examples on real-world networking tasks. It covers the advanced topics of network programming in Python using a set of selected recipes. If you are a network programmer, system/network administrator, or a web application developer, this book is ideal for you. You should have a basic familiarity with the Python programming language and TCP/IP networking concepts. However if you are a novice, you will develop an understanding of the concepts as you progress with this book. This book will serve as a supplementary material for developing hands-on skills in any academic course on network programming.

Explore various Rust features, data structures, libraries, and toolchain to build modern systems software with the help of hands-on examples Key FeaturesLearn techniques to design and build system tools and utilities in RustExplore the different features of the Rust standard library for interacting with operating systemsGain an in-depth understanding of the Rust programming language by writing low-level softwareBook Description Modern programming languages such as Python, JavaScript, and Java have become increasingly accepted for application-level programming, but for systems programming, C and C++ are predominantly used due to the need for low-level control of system resources. Rust promises the best of both worlds: the type safety of Java, and the speed and expressiveness of C++, while also including memory safety without a garbage collector. This book is a comprehensive introduction if you're new to Rust and systems programming and are looking to build reliable and efficient systems software without C or C++. The book takes a unique approach by starting each topic with Linux kernel concepts and APIs relevant to that topic. You'll also explore how system resources can be controlled from Rust. As you progress, you'll delve into advanced topics. You'll cover network programming, focusing on aspects such as working with low-level network primitives and protocols in Rust, before going on to learn how to use and compile Rust with WebAssembly. Later chapters will take you through practical code examples and projects to help you build on your knowledge. By the end of this Rust programming book, you will be equipped with practical skills to write systems software tools, libraries, and utilities in Rust. What you will learnGain a solid understanding of how system resources are managedUse Rust confidently to control and operate a Linux or Unix systemUnderstand how to write a host of practical systems software tools and utilitiesDelve into memory management with the memory layout of Rust programsDiscover the capabilities and features of the Rust Standard LibraryExplore external crates to improve productivity for future Rust programming projectsWho this book is for This book is for developers with basic knowledge of Rust but little to no knowledge or experience of systems programming. System programmers who want to consider Rust as an alternative to C or C++ will also find this book useful.

Over 60 recipes that harness the power of the Raspberry Pi together with Python programming and create enthralling and captivating projects About This Book Install your first operating system, share files over the network, and run programs remotely Construct robots and interface with your own circuits and purpose built add-ons, as well as adapt off-the-shelf household devices using this pragmatic guide Packed with clear, step-by-step recipes to walk you through the capabilities of Raspberry Pi Who This Book Is For Readers are expected to be familiar with programming concepts and Python (where possible Python 3 is used), although beginners should manage with the help of a good Python reference book and background reading. No prior knowledge of the Raspberry Pi or electronics is required; however, for the hardware sections you will need some basic electronic components/household tools to build some of the projects. What You Will Learn Get the Raspberry Pi set up and running for the first time Remotely connect to the Raspberry Pi and use your PC/laptop instead of a separate screen/keyboard Get to grips with text, files and creating quick menus using Python Develop desktop applications; handle images and process files with ease Make use of graphics and user control to develop your own exciting games Use the Raspberry Pi's powerful GPU to create 3D worlds Take control of the real world and interface with physical hardware, combining hardware and software for your own needs Measure and control processes, respond to real events and monitor through the Internet Learn about the Raspberry Pi hardware inputs/outputs, starting with the basics and beyond Expand the capabilities of the Raspberry Pi with hardware expansion / add-on modules (use analogue inputs, drive servos and motors, and use SPI/I2C) Create your own Pi-Rover or Pi-Hexpod driven by the Raspberry Pi Make use of existing hardware by modifying and interfacing with it using the Raspberry Pi In Detail Raspberry Pi cookbook for Python Programmers is a practical guide for getting the most out of this little computer. This book begins by guiding you through setting up the Raspberry Pi, performing tasks using Python 3 and introduces the first steps to interface with electronics. As you work through each chapter you will build up your skills and knowledge and apply them as you progress throughout the book, delving further and further into the unique abilities and features of the Raspberry Pi. Later, you will learn how to automate tasks by accessing files, build applications using the popular Tkinter library and create games by controlling graphics on screen. You will harness the power of the built-in graphics processor by using Pi3D to generate your own high quality 3D graphics and environments. Connect directly to the Raspberry Pi's hardware pins to control electronics from switching on LEDs and responding to push buttons right through to driving motors and servos. Learn how to monitor sensors to gather real life data and to use it to control other devices, and view the results over the Internet. Apply what you have learnt by creating your own Pi-Rover or Pi-Hexipod robots. Finally, we will explore using many of the purpose built add-ons available for the Raspberry Pi, as well as interfacing with common household devices in new ways. Style and approach Written in a cookbook style, the book contains a series of recipes on various topics, ranging from simple to complex. It is an easy-to-follow and step-by-step guide with examples of various feature integration suitable for any search application.

Secure Programming Cookbook for C and C++
Build and Design Your Own Robots
OpenCV 4 Computer Vision Application Programming Cookbook
PYTHON GUI PROJECTS WITH MACHINE LEARNING AND DEEP LEARNING
Unleash the potential of Raspberry Pi 3 with over 100 recipes, 3rd Edition
SQL Cookbook

Practical solutions to overcome challenges in creating console and web applications and working with systems-level and embedded code, network programming, deep neural networks, and much more. Key FeaturesWork through recipes featuring advanced concepts such as concurrency, unsafe code, and macros to migrate your codebase to the Rust programming language Learn how to run machine learning models with Rust Explore error handling, macros, and modularization to write maintainable codeBook Description Rust 2018, Rust's first major milestone since version 1.0, brings more advancement in the Rust language. The Rust Programming Cookbook is a practical guide to help you overcome challenges when writing Rust code. This Rust book covers recipes for configuring Rust for different environments and architectural designs, and provides solutions to practical problems. It will also take you through Rust's core concepts, enabling you to create efficient, high-performance applications that use features such as zero-cost abstractions and improved memory management. As you progress, you'll delve into more advanced topics, including channels and actors, for building scalable, production-grade applications, and even get to grips with error handling, macros, and modularization to write maintainable code. You will then learn how to overcome common roadblocks when using Rust for systems programming, IoT, web development, and network programming. Finally, you'll discover what Rust 2018 has to offer for embedded programmers. By the end of the book, you'll have learned how to build fast and safe applications and services using Rust. What you will learnUnderstand how Rust provides unique solutions to solve system programming language problemsGrasp the core concepts of Rust to develop fast and safe applicationsExplore the possibility of integrating Rust units into existing applications for improved efficiencyDiscover how to achieve better parallelism and security with RustWrite Python extensions in RustCompile external assembly files and use the Foreign Function Interface (FFI)Build web applications and services using Rust for high performanceWho this book is for The Rust cookbook is for software developers looking to enhance their knowledge of Rust and leverage its features using modern programming practices. Familiarity with Rust language is expected to get the most out of this book.

A problem-solution-based guide to help you overcome hurdles effectively while working with kernel APIs, filesystems, networks, threads, and process communications Key FeaturesLearn to apply the latest C++ features (from C++11, 14, 17, and 20) to facilitate systems programmingCreate robust and concurrent systems that make the most of the available hardware resourcesDelve into C++ inbuilt libraries and frameworks to design robust systems as per your business needsBook Description C++ is the preferred language for system programming due to its efficient low-level computation, data abstraction, and object-oriented features. System programming is about designing and writing computer programs that interact closely with the underlying operating system and allow computer hardware to interface with the programmer and the user. The C++ System Programming Cookbook will serve as a reference for developers who want to have ready-to-use solutions for the essential aspects of system programming using the latest C++ standards wherever possible. This C++ book starts out by giving you an overview of system programming and refreshing your C++ knowledge. Moving ahead, you will learn how to deal with threads and processes, before going on to discover recipes for how to manage memory. The concluding chapters will then help you understand how processes communicate and how to interact with the console (console I/O). Finally, you will learn how to deal with time interfaces, signals, and CPU scheduling. By the end of the book, you will become adept at developing robust systems applications using C++. What you will learnGet up to speed with the fundamentals including makefile, man pages, compilation, and linking and debuggingUnderstand how to deal with time interfaces, signals, and CPU schedulingDevelop your knowledge of memory managementUse processes and threads for advanced synchronizations (mutexes and condition variables)Understand interprocess communications (IPC): pipes, FIFOs, message queues, shared memory, and TCP and UDPDiscover how to interact with the console (console I/O)Who this book is for This book is for C++ developers who want to gain practical knowledge of systems programming. Though no experience of Linux system programming is assumed, intermediate knowledge of C++ is necessary.

In the last few decades, many programming languages have been developed, and there are only some that have stuck around. Some examples are C, which is a popular server development and operating system for embedded systems. When it comes to databases, the Structured Query Language (SQL) has been around since the 1970s. You can use SQL to create, generate, manage and manipulate from relational databases. Most businesses prefer to use a relational database since it can store hundreds and thousands of rows of data. This is only when the database is designed well. SQL is the only database language that can be used to manage large databases. New languages cannot compete with SQL for this reason. Hence, it is important you learn to work with SQL, and also learn how you should manage data in SQL. In this book, you will gather information about what SQL is and why it is important to learn SQL. This book also covers some of the basic commands that are used in SQL and explains how you can use those commands to manipulate information in tables and datasets. This book covers information on different data types, operators, and functions you can use to work with data and analyze data. There are many examples given across the book that will help you grasp a good understanding of what SQL is. Some exercises are also given in the book, which will help you practice some of the concepts you have learned in the book. You should continue to practice if you want to master SQL. It is okay not to know what code to use when you start learning to code in a language. It is only when you practice that you will know where you should apply a specific operator or function. Renewable energy remains one of the most important topics for a sustainable future. Wind, being a perennial source of power, could be utilized to satisfy our power requirements. With the rise of wind farms, wind power forecasting would prove to be quite useful. It contains various weather, turbine and rotor features. Data has been recorded from January 2018 till March 2020. Readings have been recorded at a 10-minute interval. The aim is to predict the wind power that could be generated from the windmill for the next 15 days. A longterm wind forecasting technique is thus required. The attributes in the dataset are as follows: ActivePower, AmbientTemperature, BearingShaftTemperature, Blade1PitchAngle, Blade2PitchAngle, Blade3PitchAngle, ControlBoxTemperature, GearboxBearingTemperature, GearboxOilTemperature, GeneratorRP, GeneratorWinding1Temperature, GeneratorWinding2Temperature, HubTemperature, MainBoxTemperature, NacellePosition, ReactivePower, RotorRPM, TurbineStatus, WTG, WindDirection, and WindSpeed. To perform forecasting based on regression active power, you will use: Linear Regression, Random Forest regression, Decision Tree regression, Support Vector Machine regression, Naïve Bayes regression, K-Nearest Neighbor regression, Adaboost regression, Gradient Boosting regression, Extreme Gradient Boosting regression, Light Gradient Boosting regression, Catboost regression, and MLP regression. To perform clustering, you will use K-Means algorithm. The machine learning models used predict categorized active power as target variable are K-Nearest Neighbor classifier, Random Forest classifier, Naive Bayes classifier, Logistic Regression classifier, Decision Tree classifier, Support Vector Machine classifier, LGBM classifier, Gradient Boosting classifier, XGB classifier, and MLP classifier. Finally, you will plot boundary decision, distribution of features, feature importance, cross validation score, and predicted values versus true values, confusion matrix, learning curve, performance of the model, scalability of the model, training loss, and training accuracy.

Raspberry Pi for Python Programmers Cookbook
Computer Organization and Assembly Language Programming
Robot Builder's Cookbook
Qt5 C++ GUI Programming Cookbook
The Art of Assembly Language, 2nd Edition
The Bulgarian C# Book

Discover interesting recipes to help you understand the concepts of object detection, image processing, and facial detection Key Features Explore the latest features and APIs in OpenCV 4 and build computer vision algorithms Develop effective, robust, and fail-safe vision for your applications Build computer vision algorithms with machine learning capabilities Book Description OpenCV is an image and video processing library used for all types of image and video analysis. Throughout the book, you'll work through recipes that implement a variety of tasks, such as facial recognition and detection. With 70 self-contained tutorials, this book examines common pain points and best practices for computer vision (CV) developers. Each recipe addresses a specific problem and offers a proven, best-practice solution with insights into how it works, so that you can copy the code and configuration files and modify them to suit your needs. This book begins by setting up OpenCV, and explains how to manipulate pixels. You'll understand how you can process images with classes and count pixels with histograms. You'll also learn detecting, describing, and matching interest points. As you advance through the chapters, you'll get to grips with estimating projective relations in images, reconstructing 3D scenes, processing video sequences, and tracking visual motion. In the final chapters, you'll cover deep learning concepts such as face and object detection. By the end of the book, you'll be able to confidently implement a range to computer vision algorithms to meet the technical requirements of your complex CV projects What you will learn Install and create a program using the OpenCV library Segment images into homogenous regions and extract meaningful objects Apply image filters to enhance image content Exploit image geometry to relay different views of a pictured scene Calibrate the camera from different image observations Detect people and objects in images using machine learning techniques Reconstruct a 3D scene from images Explore face detection using deep learning Who this book is for If you're a CV developer or professional who already uses or would like to use OpenCV for building computer vision software, this book is for you. You'll also find this book useful if you're a C++ programmer looking to extend your computer vision skillset by learning OpenCV.

Covers addressing concepts, working registers, system buses, hex dumps, machine and assembly language programming, flowcharting, microcomputer circuits, and problem-solving approaches

Build, secure, and deploy real-world serverless applications in AWS and peek into the serverless cloud offerings from Azure, Google Cloud, and IBM Cloud Key FeaturesBuild serverless applications with AWS Lambda, AWS CloudFormation and AWS CloudWatchPerform data analytics and natural language processing(NLP)on the AWS serverless platformExplore various design patterns and best practices involved in serverless computingBook Description Managing physical servers will be a thing of the past once you're able to harness the power of serverless computing. If you're already prepped with the basics of serverless computing, Serverless Programming Cookbook will help you take the next step ahead. This recipe-based guide provides solutions to problems you might face while building serverless applications. You'll begin by setting up Amazon Web Services (AWS), the primary cloud provider used for most recipes. The next set of recipes will cover various components to build a Serverless application including REST APIs, database, user management, authentication, web hosting, domain registration, DNS management, CDN, messaging, notifications and monitoring. The book also introduces you to the latest technology trends such as Data Streams, Machine Learning and NLP. You will also learn patterns and practices for using various services in a real world application. Finally, to broaden your understanding of Serverless computing, you'll also get started guides for other cloud providers such as Azure, Google Cloud Platform and IBM cloud. By the end of this book, you'll have acquired the skills you need to build serverless applications efficiently using various cloud offerings. What you will learnServerless computing in AWS and explore services with other cloudsDevelop full-stack apps with API Gateway, Cognito, Lambda and DynamoDBWeb hosting with S3, CloudFront, Route 53 and AWS Certificate ManagerSQS and SNS for effective communication between microservices Monitoring and troubleshooting with CloudWatch logs and metrics Explore Kinesis Streams, Amazon ML models and Alexa Skills KitWho this book is for For developers looking for practical solutions to common problems while building a serverless application, this book provides helpful recipes. To get started with this intermediate-level book, knowledge of basic programming is a must.

Owen Bishop introduces, through hands-on project work, the mechanics, electronics and programming involved in practical robot design-and-build. The use of the PIC microcontroller throughout provides a painless introduction to programming whilst harnessing the power of a highly popular microcontroller used by students and design engineers worldwide. This is a book for first-time robot builders, advanced builders wanting to know more about programming robots and students in Further and Higher Education tackling microcontroller-based practical work. They will all find this book a unique and exciting source of projects, ideas and techniques, to be combined into a wide range of fascinating robots. · Full step-by-step instructions for 5 complete self-build robots · Introduces key techniques in electronics, programming and construction - for robust robots that work first time · Illustrations, close-up photographs and a lively, readable text make this a fun and informative guide for novice and experienced robot builders

Become an expert C++ programmer by mastering concepts like templates, concurrency, and type deduction

Practical System Programming for Rust Developers
Python Parallel Programming Cookbook
Build complex computer vision applications with OpenCV and C++, 4th Edition
Practical solutions to building serverless applications using Java and AWS

Discover Android programming and web development by understanding the concepts of Kotlin Programming Key Features Practical solutions to your common programming problems with Kotlin 1.1 Leverage the functional power of Kotlin to ease your Android application development Learn to use Java code in conjunction with Kotlin Book Description The Android team has announced first-class support for Kotlin 1.1. This acts as an added boost to the language and more and more developers are now looking at Kotlin for their application development. This recipe-based book will be your guide to learning the Kotlin programming language. The recipes in this book build from simple language concepts to more complex applications of the language. After the fundamentals of the language, you will learn how to apply the object-oriented programming features of Kotlin 1.1. Programming with Lambdas will show you how to use the functional power of

Kotlin. This book has recipes that will get you started with Android programming with Kotlin 1.1, providing quick solutions to common problems encountered during Android app development. You will also be taken through recipes that will teach you microservice and concurrent programming with Kotlin. Going forward, you will learn to test and secure your applications with Kotlin. Finally, this book supplies recipes that will help you migrate your Java code to Kotlin and will help ensure that it's interoperable with Java. What you will learn Understand the basics and object-oriented concepts of Kotlin Programming Explore the full potential of collection frameworks in Kotlin Work with SQLite databases in Android, make network calls, and fetch data over a network Use Kotlin's Anko library for efficient and quick Android development Uncover some of the best features of Kotlin: Lambdas and Delegates Set up web service development environments, write servlets, and build RESTful services with Kotlin Learn how to write unit tests, integration tests, and instrumentation/acceptance tests. Who this book is for This book will appeal to Kotlin developers keen to find solutions for their common programming problems. Java programming knowledge would be an added advantage.

A recipe-based guide to programming your Raspberry Pi 3 using Python Key Features Leverage the power of Raspberry Pi 3 using Python programming Create 3D games, build neural network modules, and interface with your own circuits Packed with clear, step-by-step recipes to walk you through the capabilities of Raspberry Pi Book Description Raspberry Pi Cookbook for Python Programmers – Third Edition begins by guiding you through setting up Raspberry Pi 3, performing tasks using Python 3.6, and introducing the first steps to interface with electronics. As you work through each chapter, you will build your skills and apply them as you progress. You will learn how to build text classifiers, predict sentiments in words, develop applications using the popular Tkinter library, and create games by controlling graphics on your screen. You will harness the power of a built in graphics processor using Pi3D to generate your own high-quality 3D graphics and environments. You will understand how to connect Raspberry Pi's hardware pins directly to control electronics, from switching on LEDs and responding to push buttons to driving motors and servos. Get to grips with monitoring sensors to gather real-life data, using it to control other devices, and viewing the results over the internet. You will apply what you have learned by creating your own Pi-Rover or Pi-Hexipod robots. You will also learn about sentiment analysis, face recognition techniques, and building neural network modules for optical character recognition. Finally, you will learn to build movie recommendations system on Raspberry Pi 3. What you will learn Learn to set up and run Raspberry Pi 3 Build text classifiers and perform automation using Python Predict sentiments in words and create games and graphics Detect edges and contours in images Build human face detection and recognition system Use Python to drive hardware Sense and display real-world data Build a neural network module for optical character recognition Build movie recommendations system Who this book is for This book is for anyone who wants to master the skills of Python programming using Raspberry Pi 3. Prior knowledge of Python will be an added advantage.

This is a cookbook that shows results obtained on real images with detailed explanations and the relevant screenshots. The recipes contain code accompanied with suitable explanations that will facilitate your learning. If you are a novice C++ programmer who wants to learn how to use the OpenCV library to build computer vision applications, then this cookbook is appropriate for you. It is also suitable for professional software developers wishing to be introduced to the concepts of computer vision programming. It can be used as a companion book in university-level computer vision courses. It constitutes an excellent reference for graduate students and researchers in image processing and computer vision. The book provides a good combination of basic to advanced recipes. Basic knowledge of C++ is required.

A recipe-based guide to refining your C++ programming skills with the help of coding best practices, advanced programming concepts, and the latest features of C++17 and C++20 Key FeaturesLearn how to develop and design your own librariesFind solutions to your app development problems and implement them in a highly reusable manner, following library development best practicesExplore advanced C++ features such as containers, coroutines, and modulesBook Description If you think you've mastered C++ and know everything it takes to write robust applications, you'll be in for a surprise. With this book, you'll gain comprehensive insights into C++, covering exclusive tips and interesting techniques to enhance your app development process. You'll kick off with the basic principles of library design and development, which will help you understand how to write reusable and maintainable code. You'll then discover the importance of exception safety, and how you can avoid unexpected errors or bugs in your code. The book will take you through the modern elements of C++, such as move semantics, type deductions, and coroutines. As you advance, you'll delve into template programming – the standard tool for most library developers looking to achieve high code reusability. You'll explore the STL and learn how to avoid common pitfalls while implementing templates. Later, you'll learn about the problems of multithreaded programming such as data races, deadlocks, and thread starvation. You'll also learn high-performance programming by using benchmarking tools and libraries. Finally, you'll discover advanced techniques for debugging and testing to ensure code reliability. By the end of this book, you'll have become an expert at C++ programming and will have gained the skills to solve complex development problems with ease. What you will learnSolve common C++ development problems by implementing solutions in a more generic and reusable wayAchieve different levels of exception safety guarantees by introducing precise declarationsWrite library-quality code that meets professional standardsPractice writing reliable, performant code that exposes consistent behavior in programsUnderstand why you need to implement design patterns and how it's doneWork with complex examples to understand various aspects of good library designWho this book is for This book is for intermediate and expert-level C++ developers who are looking to explore the lesser known functionalities of the language to improve the efficiency of their code and the way they develop applications. Basic knowledge of object-oriented programming concepts and the Standard Template Library (STL) is assumed.

Explore the latest features of Rust 2018 for building fast and secure apps
Build fast and secure software for Linux/Unix systems with the help of practical examples
WIND POWER ANALYSIS AND FORECASTING USING MACHINE LEARNING WITH PYTHON
ARM® Cortex® M4 Cookbook
Python Cookbook
Go Programming Cookbook

The Chaos Cookbook: A Practical Programming Guide discusses the use of chaos in computer programming. The book is comprised of 11 chapters that tackle various topics relevant to chaos and programming. Chapter 1 reviews the concept of chaos, and Chapter 2 discusses the iterative functions. Chapters 3 and 4 cover differential and Lorenz equations. Chapter 5 talks about strange attractors, while Chapter 6 deals with the fractal link. The book also discusses the Mandelbrot set, and then covers the Julia sets. The other fractal systems and the cellular automata are also explained. The last chapter discusses practical chaos. The book will be of great use to professionals, students, and hobbyist programmers who have an interest with the chaos systems.

Overcome the vexing issues you're likely to face when creating apps for the iPhone, iPad, or iPod touch. With new and thoroughly revised recipes in this updated cookbook, you'll quickly learn the steps necessary to work with the iOS 7 SDK--including ways to store and protect data, send and receive notifications, enhance and animate graphics, manage files and folders, and take advantage of UI Dynamics.

The free book "Fundamentals of Computer Programming with C#" is a comprehensive computer programming tutorial that teaches programming, logical thinking, data structures and algorithms, problem solving and high quality code with lots of examples in C#. It starts with the first steps in programming and software development like variables, data types, conditional statements, loops and arrays and continues with other basic topics like methods, numeral systems, strings and string processing, exceptions, classes and objects. After the basics this fundamental programming book enters into more advanced programming topics like recursion, data structures (lists, trees, hash-tables and graphs), high-quality code, unit testing and refactoring, object-oriented principles (inheritance, abstraction, encapsulation and polymorphism) and their implementation the C# language. It also covers fundamental topics that each good developer should know like algorithm design, complexity of algorithms and problem solving. The book uses C# language and Visual Studio to illustrate the programming concepts and explains some C# / .NET specific technologies like lambda expressions, extension methods and LINQ. The book is written by a team of developers lead by Svetlin Nakov who has 20+ years practical software development experience. It teaches the major programming concepts and way of thinking needed to become a good software engineer and the C# language in the meantime. It is a great start for anyone who wants to become a skillful software engineer. The books does not teach technologies like databases, mobile and web development, but shows the true way to master the basics of programming regardless of the languages, technologies and tools. It is good for beginners and intermediate developers who want to put a solid base for a successful career in the software engineering industry. The book is accompanied by free video lessons, presentation slides and mind maps, as well as hundreds of exercises and live examples. Download the free C# programming book, videos, presentations and other resources from http://introprogramming.info. Title: Fundamentals of Computer Programming with C# (The Bulgarian C# Programming Book) ISBN: 9789544007737 ISBN-13: 978-954-400-773-7 (9789544007737) ISBN-10: 954-400-773-3 (9544007733) Author: Svetlin Nakov & Co. Pages: 1132 Language: English Published: Sofia, 2013 Publisher: Faber Publishing, Bulgaria Web site: http://www.introprogramming.info License: CC-Attribution-Share-Alike Tags: free, programming, book, computer programming, programming fundamentals, ebook, book programming, C#, CSharp, C# book, tutorial, C# tutorial; programming concepts, programming fundamentals, compiler, Visual Studio, .NET, .NET Framework, data types, variables, expressions, statements, console, conditional statements, control-flow logic, loops, arrays, numeral systems, methods, strings, text processing, StringBuilder, exceptions, exception handling, stack trace, streams, files, text files, linear data structures, list, linked list, stack, queue, tree, balanced tree, graph, depth-first search, DFS, breadth-first search, BFS, dictionaries, hash tables, associative arrays, sets, algorithms, sorting algorithm, searching algorithms, recursion, combinatorial algorithms, algorithm complexity, OOP, object-oriented programming, classes, objects, constructors, fields, properties, static members, abstraction, interfaces, encapsulation, inheritance, virtual methods, polymorphism, cohesion, coupling, enumerations, generics, namespaces, UML, design patterns, extension methods, anonymous types, lambda expressions, LINQ, code quality, high-quality code, high-quality classes, high-quality methods, code formatting, self-documenting code, code refactoring, problem solving, problem solving methodology, 9789544007737, 9544007733

Discover interesting recipes to help you understand the concepts of object detection, image processing, and facial detection Key Features Explore the latest features and APIs in OpenCV 4 and build computer vision algorithms Develop effective, robust, and fail-safe vision for your applications Build computer vision algorithms with machine learning capabilities Book Description OpenCV is an image and video processing library used for all types of image and video analysis. Throughout the book, you'll work through recipes that implement a variety of tasks. With 70 self-contained tutorials, this book examines common pain points and best practices for computer vision (CV) developers. Each recipe addresses a specific problem and offers a proven, best-practice solution with insights into how it works, so that you can copy the code and configuration files and modify them to suit your needs. This book begins by setting up OpenCV, and explains how to manipulate pixels. You'll understand how you can process images with classes and count pixels with histograms. You'll also learn detecting, describing, and matching interest points. As you advance through the chapters, you'll get to grips with estimating projective relations in images, reconstructing 3D scenes, processing video sequences, and tracking visual motion. In the final chapters, you'll cover deep learning concepts such as face and object detection. By the end of the book, you'll be able to confidently implement a range of computer vision algorithms to meet the technical requirements of your complex CV projects. What you will learn Install and create a program using the OpenCV library Segment images into homogenous regions and extract meaningful objects Apply image filters to enhance image content Exploit image geometry to relay different views of a pictured scene Calibrate the camera from different image observations Detect people and objects in images using machine learning techniques Reconstruct a 3D scene from images Explore face detection using deep learning Who this book is for If you're a CV developer or professional who already uses or would like to use OpenCV for building computer vision software, this book is for you. You'll also find this book useful if you're a C++ programmer looking to extend your computer vision skillset by learning OpenCV. Downloading the example code for this ebook: You can download the example code files for this ebook on GitHub at the following link: https://github ...

Over 50 Recipes to Master this Library of Programming Functions for Real-time Computer Vision
OpenCV 3 Computer Vision Application Programming Cookbook
How to Quickly Learn Structured Query Language Programming, Server Administration, Computer and Database Management Step-by-Step
IOS 7 Programming Cookbook
Recipes for Mastering Python 3
OpenCV 4 Computer Vision Application Programming Cookbook - Fourth Edition

Recipes to help you build computer vision applications that make the most of the popular C++ library OpenCV 3 About This Book Written to the latest, gold-standard specification of OpenCV 3 Master OpenCV, the open source library of the computer vision community Master fundamental concepts in computer vision and image processing Learn about the important classes and functions of OpenCV with complete working examples applied to real images Who This Book Is For OpenCV 3 Computer Vision Application Programming Cookbook Third Edition is appropriate for novice C++ programmers who want to learn how to use the OpenCV library to build computer vision applications. It is also suitable for professional software developers who wish to be introduced to the concepts of computer vision programming. It can also be used as a companion book in university-level computer vision courses. It constitutes an excellent reference for graduate students and researchers in image processing and computer vision. What You Will Learn Install and create a program using the OpenCV library Process an image by manipulating its pixels Analyze an image using histograms Segment images into homogenous regions and extract meaningful objects Apply image filters to enhance image content Exploit the image geometry in order to relay different views of a pictured scene Calibrate the camera from different image observations Detect people and objects in images using machine learning techniques Reconstruct a 3D scene from images In Detail Making your applications see has never been easier with OpenCV. With it, you can teach your robot how to follow your cat, write a program to correctly identify the members of One Direction, or even help you find the right colors for your redecoration. OpenCV 3 Computer Vision Application Programming Cookbook Third Edition provides a complete introduction to the OpenCV library and explains how to build your first computer vision program. You will be presented with a variety of computer vision algorithms and exposed to important concepts in image and video analysis that will enable you to build your own computer vision applications. This book helps you to get started with the library, and shows you how to install and deploy the OpenCV library to write effective computer vision applications following good programming practices. You will learn how to read and write images and manipulate their pixels. Different techniques for image enhancement and shape analysis will be presented. You will learn how to detect specific image features such as lines, circles or corners. You will be introduced to the concepts of mathematical morphology and image filtering. The most recent methods for image matching and object recognition are described, and you'll discover how to process video from files or cameras, as well as how to detect and track moving objects. Techniques to achieve camera calibration and perform multiple-view analysis will also be explained. Finally, you'll also get acquainted with recent approaches in machine learning and object classification. Style and approach This book will arm you with the basics you need to start writing world-aware applications right from a pixel level all the way through to processing video sequences.

Master efficient parallel programming to build powerful applications using Python About This Book Design and implement efficient parallel software Master new programming techniques to address and solve complex programming problems Explore the world of parallel programming with this book, which is a go-to resource for different kinds of parallel computing tasks in Python, using examples and topics covered in great depth Who This Book Is For Python Parallel Programming Cookbook is intended for software developers who are well versed with Python and want to use parallel programming techniques to write powerful and efficient code. This book will help you master the basics and the advanced of parallel computing. What You Will Learn Synchronize multiple threads and processes to manage parallel tasks Implement message passing communication between processes to build parallel applications Program your own GPU cards to address complex problems Manage computing entities to execute distributed computational tasks Write efficient programs by adopting the event-driven programming model Explore the cloud technology with DJango and Google App Engine Apply parallel programming techniques that can lead to performance improvements In Detail Parallel programming techniques are required for a developer to get the best use of all the computational resources available today and to build efficient software systems. From multi-core to GPU systems up to the distributed architectures, the high computation of programs throughout requires the use of programming tools and software libraries. Because of this, it is becoming increasingly important to know what the parallel programming techniques are. Python is commonly used as even non-experts can easily deal with its concepts. This book will teach you parallel programming techniques using examples in Python and will help you explore the many ways in which you can write code that allows more than one process to happen at once. Starting with introducing you to the world of parallel computing, it moves on to cover the fundamentals in Python. This is followed by exploring the thread-based parallelism model using the Python threading module by synchronizing threads and using locks, mutex, semaphores queues, GIL, and the thread pool. Next you will be taught about process-based parallelism where you will synchronize processes using message passing along with learning about the performance of MPI Python Modules. You will then go on to learn the asynchronous parallel programming model using the Python asyncio module along with handling exceptions. Moving on, you will discover distributed computing with Python, and learn how to install a broker, use Celery Python Module, and create a worker. You will also understand the StarCluster framework, Pycsp, Scoop, and Disco modules in Python. Further on, you will learn GPU programming with Python using the PyCUDA module along with evaluating performance limitations. Next you will get acquainted with the cloud computing concepts in Python, using Google App Engine (GAE), and building your first application with GAE. Lastly, you will learn about grid computing concepts in Python and using PyGlobus toolkit, GFTP and GASS COPY to transfer files, and service monitoring in PyGlobus. Style and approach A step-by-step guide to parallel programming using Python, with recipes accompanied by one or more programming examples. It is a practically oriented book and has all the necessary underlying parallel computing concepts.

Discover the new features and widely used packages in Julia to solve complex computational problems in your statistical applications. Key FeaturesAddress the core problems of programming in Julia with the most popular packages for common tasksTackle issues while working with Databases and Parallel data processing with JuliaExplore advanced features such as metaprogramming, functional programming, and user defined typesBook Description Julia, with its dynamic nature and high-performance, provides comparatively minimal time for the development of computational models with easy-to-maintain computational code. This book will be your solution-based guide as it will take you through different programming aspects with Julia. Starting with the new features of Julia 1.0, each recipe addresses a specific problem, providing a solution and explaining how it works. You will work with the powerful Julia tools and data structures along with the most popular Julia packages. You will learn to create vectors, handle variables, and work with functions. You will be introduced to various recipes for numerical computing, distributed computing, and achieving high performance. You will see how to optimize data science programs with parallel computing and memory allocation. We will look into more advanced concepts such as metaprogramming and functional programming. Finally, you will learn how to tackle issues while working with databases and data processing, and will learn about data science problems, data modeling, data analysis, data manipulation, parallel processing, and cloud computing with Julia. By the end of the book, you will have acquired the skills to work more effectively with your data What you will learnBoost your code's performance using Julia's unique featuresOrganize data in fundamental types of collections: arrays and dictionariesOrganize data science processes within Julia and solve related problemsScale Julia computations with cloud computingWrite data to IO streams with Julia and handle web transferDefine your own immutable and mutable typesSpeed up the development process using metaprogrammingWho this book is for This book is for developers who would like to enhance their Julia programming skills and would like to get some quick solutions to their common programming problems. Basic Julia programming knowledge is assumed.

Use Qt 5 to design and build functional, appealing, and user-friendly graphical user interfaces (GUIs) for your applications. Key FeaturesLearn to design and customize the look and feel of your applicationImprove the visual quality of an application by using graphics rendering and animationUnderstand the balance of presentation and web content that will make an application appealing yet functionalBook Description With the growing need to develop GUIs for multiple targets and multiple screens, improving the visual quality of your application becomes important so that it stands out from your competitors. With its cross-platform ability and the latest UI paradigms, Qt makes it possible to build intuitive, interactive, and user-friendly user interfaces for your applications. Qt5 C++ GUI Programming Cookbook, Second Edition teaches you how to develop functional and appealing user interfaces using the latest version of QT5 and C++. This book will help you learn a variety of topics such as GUI customization and animation, graphics rendering, implementing Google Maps, and more. You will also be taken through advanced concepts like asynchronous programming, event handling using signals and slots, network programming, various aspects of optimizing your application. By the end of the book, you will be confident to design and customize GUI applications that meet your clients' expectations and have an understanding of best practice solutions for common problems. What you will learnAnimate GUI elements using Qt5's built-in animation systemDraw shapes and 2D images using Qt5's powerful rendering systemImplement an industry-standard OpenGL library in your projectBuild a mobile app that supports touch events and exports it onto devicesParse and extract data from an XML file and present it on your GUIInteract with web content by calling JavaScript functions from C++Access MySQL and SQLite databases to retrieve data and display it on your GUIWho this book is for This intermediate-level book is designed for those who want to develop software using Qt 5. If you want to improve the visual quality and content presentation of your software application, this book is for you. Prior experience of C++ programming is required.

Clojure Programming Cookbook
A Practical Programming Guide
Don Lancaster's Micro Cookbook: Fundamentals
Explore more than 100 recipes that show how to build robust mobile and web applications with Kotlin, Spring Boot, and Android
Recipes for Cryptography, Authentication, Input Validation & More
Fundamentals of Computer Programming with C#

Tackle the trickiest of problems in Go programming with this practical guide Key Features Develop applications for different domains using modern programming techniques Tackle common problems when it comes to parallelism, concurrency, and reactive programming in Go Work with ready-to-execute code based on the latest version of Go Book Description Go (or Golang) is a statically typed programming language developed at Google. Known for its vast standard library, it also provides features such as garbage collection, type safety, dynamic-typing capabilities, and additional built-in types. This book will serve as a reference while implementing Go features in your own applications. This Go cookbook helps you put into practice the advanced concepts and libraries that Golang offers. The recipes in the book follow best practices such as documentation, testing, and vendoring with Go modules, as well as performing clean abstractions using interfaces. You'll learn how code works and the common pitfalls to watch out for. The book covers basic type and error handling, and then moves on to explore applications, such as websites, command-line tools, and filesystems, that interact with users. You'll even get to grips with parallelism, distributed systems, and performance tuning. By the end of the

book, you'll be able to use open source code and concepts in Go programming to build enterprise-class applications without any hassle. What you will learn Work with third-party Go projects and modify them for your use Write Go code using modern best practices Manage your dependencies with the new Go module system Solve common problems encountered when dealing with backend systems or DevOps Explore the Go standard library and its uses Test, profile, and fine-tune Go applications Who this book is for If you're a web developer, programmer, or enterprise developer looking for quick solutions to common and not-so-common problems in Go programming, this book is for you. Basic knowledge of the Go language is assumed.

Raspberry Pi Cookbook for Python Programmers is written in a Cookbook format, presenting examples in the style of recipes.This allows you to go directly to your topic of interest, or follow topics throughout a chapter to gain a thorough in-depth knowledge. The aim of this book is to bring you a broad range of Python 3 examples and practical ideas which you can develop to suit your own requirements. By modifying and combining the examples to create your own projects you learn far more effectively with a much greater understanding. Each chapter is designed to become a foundation for further experimentation and discovery of the topic, providing you with the tools and information to jump right in. Readers are expected to be familiar with programming concepts and Python (where possible Python 3 is used), although beginners should manage with the help of a good Python reference book and background reading. No prior knowledge of the Raspberry Pi or electronics is required; however for the hardware sections you will need some basic electronic components/household tools to build some of the projects.

Password sniffing, spoofing, buffer overflows, and denial of service: these are only a few of the attacks on today's computer systems and networks. At the root of this epidemic is poorly written, poorly tested, and insecure code that puts everyone at risk. Clearly, today's developers need help figuring out how to write code that attackers won't be able to exploit. But writing such code is surprisingly difficult. Secure Programming Cookbook for C and C++ is an important new resource for developers serious about writing secure code. It contains a wealth of solutions to problems faced by those who care about the security of their applications. It covers a wide range of topics, including safe initialization, access control, input validation, symmetric and public key cryptography, cryptographic hashes and MACs, authentication and key exchange, PKI, random numbers, and anti-tampering. The rich set of code samples provided in the book's more than 200 recipes will help programmers secure the C and C++ programs they write for both Unix® (including Linux®) and Windows® environments. Readers will learn: How to avoid common programming errors, such as buffer overflows, race conditions, and format string problems How to properly SSL-enable applications How to create secure channels for client-server communication without SSL How to integrate Public Key Infrastructure (PKI) into applications Best practices for using cryptography properly Techniques and strategies for properly validating input to programs How to launch programs securely How to use file access mechanisms properly Techniques for protecting applications from reverse engineering The book's web site supplements the book by providing a place to post new recipes, including those written in additional languages like Perl, Java, and Python. Monthly prizes will reward the best recipes submitted by readers. Secure Programming Cookbook for C and C++ is destined to become an essential part of any developer's library, a code companion developers will turn to again and again as they seek to protect their systems from attackers and reduce the risks they face in today's dangerous world.

Why spend time on coding problems that others have already solved when you could be making real progress on your Ruby project? This updated cookbook provides more than 350 recipes for solving common problems, on topics ranging from basic data structures, classes, and objects, to web development, distributed programming, and multithreading. Revised for Ruby 2.1, each recipe includes a discussion on why and how the solution works. You'll find recipes suitable for all skill levels, from Ruby newbies to experts who need an occasional reference. With Ruby Cookbook, you'll not only save time, but keep your brain percolating with new ideas as well. Recipes cover: Data structures including strings, numbers, date and time, arrays, hashes, files and directories Using Ruby's code blocks, also known as closures OOP features such as classes, methods, objects, and modules XML and HTML, databases and persistence, and graphics and other formats Web development with Rails and Sinatra Internet services, web services, and distributed programming Software testing, debugging, packaging, and distributing Multitasking, multithreading, and extending Ruby with other languages

Julia 1.0 Programming Cookbook
C++ System Programming Cookbook
Rust Programming Cookbook
Kotlin Programming Cookbook
C# Programming Cookbook
Practical recipes for building cross-platform GUI applications, widgets, and animations with Qt 5, 2nd Edition

A recipe-based guide to refining your C++ programming skills with the help of coding best practices, advanced programming concepts, and the latest features of C++17 and C++20 Key Features Learn how to develop and design your own libraries Find solutions to your app development problems and implement them in a highly reusable manner, following library development best practices Explore advanced C++ features such as containers, coroutines, and modules Book Description If you think you've mastered C++ and know everything it takes to write robust applications, you'll be in for a surprise. With this book, you'll gain comprehensive insights into C++, covering exclusive tips and interesting techniques to enhance your app development process. You'll kick off with the basic principles of library design and development, which will help you understand how to write reusable and maintainable code. You'll then discover the importance of exception safety, and how you can avoid unexpected errors or bugs in your code. The book will take you through the modern elements of C++, such as move semantics, type deductions, and coroutines. As you advance, you'll delve into template programming - the standard tool for most library developers looking to achieve high code reusability. You'll explore the STL and learn how to avoid common pitfalls while implementing templates. Later, you'll learn about the problems of multithreaded programming such as data races, deadlocks, and thread starvation. You'll also learn high-performance programming by using benchmarking tools and libraries. Finally, you'll discover advanced techniques for debugging and testing to ensure code reliability. By the end of this book, you'll have become an expert at C++ programming and will have gained the skills to solve complex development problems with ease. What you will learn Solve common C++ development problems by implementing solutions in a more generic and reusable way Achieve different levels of exception safety guarantees by introducing precise declarations Write library-quality code that meets professional standards Practice writing reliable, performant code that exposes consistent behavior in programs Understand why you need to implement design patterns and how it's done Work with complex examples to understand various aspects of good library design Who this book is for This book is for intermediate and expert-level C++ developers who are looking to explore the lesser known functionalities of the language to improve the efficiency of their code and the way they develop applications. Basic knowledge of object-oriented programming concepts and the Standard Template Library (STL) is assumed.

Assembly is a low-level programming language that's one step above a computer's native machine language. Although assembly language is commonly used for writing device drivers, emulators, and video games, many programmers find its somewhat unfriendly syntax intimidating to learn and use. Since 1996, Randall Hyde's The Art of Assembly Language has provided a comprehensive, plain-English, and patient introduction to 32-bit x86 assembly for non-assembly programmers. Hyde's primary teaching tool, High Level Assembler (or HLA), incorporates many of the features found in high-level languages (like C, C++, and Java) to help you quickly grasp basic assembly concepts. HLA lets you write true low-level code while enjoying the benefits of high-level language programming. As you read The Art of Assembly Language, you'll learn the low-level theory fundamental to computer science and turn that understanding into real, functional code. You'll learn how to: –Edit, compile, and run HLA programs –Declare and use constants, scalar variables, pointers, arrays, structures, unions, and namespaces –Translate arithmetic expressions (integer and floating point) –Convert high-level control structures This much anticipated second edition of The Art of Assembly Language has been updated to reflect recent changes to HLA and to support Linux, Mac OS X, and FreeBSD. Whether you're new to programming or you have experience with high-level languages, The Art of Assembly Language, 2nd Edition is your essential guide to learning this complex, low-level language.

Handle every problem you come across in the world of Clojure programming with this expert collection of recipes About This Book Discover a wide variety of practical cases and real world techniques to enhance your productivity with Clojure. Learn to resolve the everyday issues you face with a functional mindset using Clojure You will learn to write highly efficient, more productive, and error-free programs without the risk of deadlocks and race-conditions Who This Book Is For This book is for Clojure developers who have some Clojure programming experience and are well aware of their shortcomings. If you want to learn to tackle common problems, become an expert, and develop a solid skill set, then this book is for you. What You Will Learn Manipulate, access, filter, and transform your data with Clojure Write efficient parallelized code through Clojure abstractions Tackle Complex Concurrency easily with Reactive Programming Build on Haskell abstractions to write dynamic functional tests Write AWS Lambda functions effortlessly Put Clojure in use into your IoT devices Use Clojure with Slack for instant monitoring Scaling your Clojure application using Docker Develop real-time system interactions using MQTT and websockets In Detail When it comes to learning and using a new language you need an effective guide to be by your side when things get rough. For Clojure developers, these recipes have everything you need to take on everything this language offers. This book is divided into three high impact sections. The first section gives you an introduction to live programming and best practices. We show you how to interact with your connections by manipulating, transforming, and merging collections. You'll learn how to work with macros, protocols, multi-methods, and transducers. We'll also teach you how to work with languages such as Java, and Scala. The next section deals with intermediate-level content and enhances your Clojure skills, here we'll teach you concurrency programming with Clojure for high performance. We will provide you with advanced best practices, tips on Clojure programming, and show you how to work with Clojure while developing applications. In the final section you will learn how to test, deploy and analyze websocket behavior when your app is deployed in the cloud. Finally, we will take you through DevOps. Developing with Clojure has never been easier with these recipes by your side! Style and approach This book takes a recipe-based approach by diving directly into helpful programming concepts. It will give you a foolproof approach to programming and teach you how to deal with problems that may arise while working with Clojure. The book is divided into three sections giving you the freedom skip to the section of your choice depending on the problem faced.

Master C++ core language and standard library features, with over 100 recipes, updated to C++20, 2nd Edition
Practical recipes for Linux system-level programming using the latest C++ features
Serverless Programming Cookbook
Modern C++ Programming Cookbook
The Chaos Cookbook
Over 100 numerical and distributed computing recipes for your daily data science workflow