1. **Chapter 1 Data Structures and Algorithms** 
   1. **Unpacking a Sequence into Separate Variables**
   2. **Unpacking Elements from Iterables of Arbitrary Length**
   3. **Keeping the Last N Items**
   4. **Finding the Largest or Smallest N Items**
   5. **Implementing a Priority Queue**
   6. **Mapping Keys to Multiple Values in a Dictionary**
   7. **Keeping Dictionaries in Order**
   8. **Calculating with Dictionaries**
   9. **Finding Commonalities in Two Dictionaries**
   10. **Removing Duplicates from a Sequence while Maintaining Order**
   11. **Naming a Slice**
   12. **Determining the Most Frequently Occurring Items in a Sequence**
   13. **Sorting a List of Dictionaries by a Common Key**
   14. **Sorting Objects Without Native Comparison Support**
   15. **Grouping Records Together Based on a Field**
   16. **Filtering Sequence Elements**
   17. **Extracting a Subset of a Dictionary**
   18. **Mapping Names to Sequence Elements**
   19. **Transforming and Reducing Data at the Same Time**
   20. **Combining Multiple Mappings into a Single Mapping**
2. **Chapter 2 Strings and Text** 
   1. **Splitting Strings on Any of Multiple Delimiters**
   2. **Matching Text at the Start or End of a String**
   3. **Matching Strings Using Shell Wildcard Patterns**
   4. **Matching and Searching for Text Patterns**
   5. **Searching and Replacing Text**
   6. **Searching and Replacing Case-Insensitive Text**
   7. **Specifying a Regular Expression for the Shortest Match**
   8. **Writing a Regular Expression for Multiline Patterns**
   9. **Normalizing Unicode Text to a Standard Representation**
   10. **Working with Unicode Characters in Regular Expressions**
   11. **Stripping Unwanted Characters from Strings**
   12. **Sanitizing and Cleaning Up Text**
   13. **Aligning Text Strings**
   14. **Combining and Concatenating Strings**
   15. **Interpolating Variables in Strings**
   16. **Reformatting Text to a Fixed Number of Columns**
   17. **Handling HTML and XML Entities in Text**
   18. **Tokenizing Text**
   19. **Writing a Simple Recursive Descent Parser**
   20. **Performing Text Operations on Byte Strings**
3. **Chapter 3 Numbers, Dates, and Times** 
   1. **Rounding Numerical Values**
   2. **Performing Accurate Decimal Calculations**
   3. **Formatting Numbers for Output**
   4. **Working with Binary, Octal, and Hexadecimal Integers**
   5. **Packing and Unpacking Large Integers from Bytes**
   6. **Performing Complex-Valued Math**
   7. **Working with Infinity and NaNs**
   8. **Calculating with Fractions**
   9. **Calculating with Large Numerical Arrays**
   10. **Performing Matrix and Linear Algebra Calculations**
   11. **Picking Things at Random**
   12. **Converting Days to Seconds, and Other Basic Time Conversions**
   13. **Determining Last Friday’s Date**
   14. **Finding the Date Range for the Current Month**
   15. **Converting Strings into Datetimes**
   16. **Manipulating Dates Involving Time Zones**
4. **Chapter 4 Iterators and Generators** 
   1. **Manually Consuming an Iterator**
   2. **Delegating Iteration**
   3. **Creating New Iteration Patterns with Generators**
   4. **Implementing the Iterator Protocol**
   5. **Iterating in Reverse**
   6. **Defining Generator Functions with Extra State**
   7. **Taking a Slice of an Iterator**
   8. **Skipping the First Part of an Iterable**
   9. **Iterating Over All Possible Combinations or Permutations**
   10. **Iterating Over the Index-Value Pairs of a Sequence**
   11. **Iterating Over Multiple Sequences Simultaneously**
   12. **Iterating on Items in Separate Containers**
   13. **Creating Data Processing Pipelines**
   14. **Flattening a Nested Sequence**
   15. **Iterating in Sorted Order Over Merged Sorted Iterables**
   16. **Replacing Infinite while Loops with an Iterator**
5. **Chapter 5 Files and I/O** 
   1. **Reading and Writing Text Data**
   2. **Printing to a File**
   3. **Printing with a Different Separator or Line Ending**
   4. **Reading and Writing Binary Data**
   5. **Writing to a File That Doesn’t Already Exist**
   6. **Performing I/O Operations on a String**
   7. **Reading and Writing Compressed Datafiles**
   8. **Iterating Over Fixed-Sized Records**
   9. **Reading Binary Data into a Mutable Buffer**
   10. **Memory Mapping Binary Files**
   11. **Manipulating Pathnames**
   12. **Testing for the Existence of a File**
   13. **Getting a Directory Listing**
   14. **Bypassing Filename Encoding**
   15. **Printing Bad Filenames**
   16. **Adding or Changing the Encoding of an Already Open File**
   17. **Writing Bytes to a Text File**
   18. **Wrapping an Existing File Descriptor As a File Object**
   19. **Making Temporary Files and Directories**
   20. **Communicating with Serial Ports**
   21. **Serializing Python Objects**
6. **Chapter 6 Data Encoding and Processing** 
   1. **Reading and Writing CSV Data**
   2. **Reading and Writing JSON Data**
   3. **Parsing Simple XML Data**
   4. **Parsing Huge XML Files Incrementally**
   5. **Turning a Dictionary into XML**
   6. **Parsing, Modifying, and Rewriting XML**
   7. **Parsing XML Documents with Namespaces**
   8. **Interacting with a Relational Database**
   9. **Decoding and Encoding Hexadecimal Digits**
   10. **Decoding and Encoding Base64**
   11. **Reading and Writing Binary Arrays of Structures**
   12. **Reading Nested and Variable-Sized Binary Structures**
   13. **Summarizing Data and Performing Statistics**
7. **Chapter 7 Functions** 
   1. **Writing Functions That Accept Any Number of Arguments**
   2. **Writing Functions That Only Accept Keyword Arguments**
   3. **Attaching Informational Metadata to Function Arguments**
   4. **Returning Multiple Values from a Function**
   5. **Defining Functions with Default Arguments**
   6. **Defining Anonymous or Inline Functions**
   7. **Capturing Variables in Anonymous Functions**
   8. **Making an N-Argument Callable Work As a Callable with Fewer Arguments**
   9. **Replacing Single Method Classes with Functions**
   10. **Carrying Extra State with Callback Functions**
   11. **Inlining Callback Functions**
   12. **Accessing Variables Defined Inside a Closure**
8. **Chapter 8 Classes and Objects** 
   1. **Changing the String Representation of Instances**
   2. **Customizing String Formatting**
   3. **Making Objects Support the Context-Management Protocol**
   4. **Saving Memory When Creating a Large Number of Instances**
   5. **Encapsulating Names in a Class**
   6. **Creating Managed Attributes**
   7. **Calling a Method on a Parent Class**
   8. **Extending a Property in a Subclass**
   9. **Creating a New Kind of Class or Instance Attribute**
   10. **Using Lazily Computed Properties**
   11. **Simplifying the Initialization of Data Structures**
   12. **Defining an Interface or Abstract Base Class**
   13. **Implementing a Data Model or Type System**
   14. **Implementing Custom Containers**
   15. **Delegating Attribute Access**
   16. **Defining More Than One Constructor in a Class**
   17. **Creating an Instance Without Invoking init**
   18. **Extending Classes with Mixins**
   19. **Implementing Stateful Objects or State Machines**
   20. **Calling a Method on an Object Given the Name As a String**
   21. **Implementing the Visitor Pattern**
   22. **Implementing the Visitor Pattern Without Recursion**
   23. **Managing Memory in Cyclic Data Structures**
   24. **Making Classes Support Comparison Operations**
   25. **Creating Cached Instances**
9. **Chapter 9 Metaprogramming** 
   1. **Putting a Wrapper Around a Function**
   2. **Preserving Function Metadata When Writing Decorators**
   3. **Unwrapping a Decorator**
   4. **Defining a Decorator That Takes Arguments**
   5. **Defining a Decorator with User Adjustable Attributes**
   6. **Defining a Decorator That Takes an Optional Argument**
   7. **Enforcing Type Checking on a Function Using a Decorator**
   8. **Defining Decorators As Part of a Class**
   9. **Defining Decorators As Classes**
   10. **Applying Decorators to Class and Static Methods**
   11. **Writing Decorators That Add Arguments to Wrapped Functions**
   12. **Using Decorators to Patch Class Definitions**
   13. **Using a Metaclass to Control Instance Creation**
   14. **Capturing Class Attribute Definition Order**
   15. **Defining a Metaclass That Takes Optional Arguments**
   16. **Enforcing an Argument Signature on \*args and \*\*kwargs**
   17. **Enforcing Coding Conventions in Classes**
   18. **Defining Classes Programmatically**
   19. **Initializing Class Members at Definition Time**
   20. **Implementing Multiple Dispatch with Function Annotations**
   21. **Avoiding Repetitive Property Methods**
   22. **Defining Context Managers the Easy Way**
   23. **Executing Code with Local Side Effects**
   24. **Parsing and Analyzing Python Source**
   25. **Disassembling Python Byte Code**
10. **Chapter 10 Modules and Packages** 
    1. **Making a Hierarchical Package of Modules**
    2. **Controlling the Import of Everything**
    3. **Importing Package Submodules Using Relative Names**
    4. **Splitting a Module into Multiple Files**
    5. **Making Separate Directories of Code Import Under a Common Namespace**
    6. **Reloading Modules**
    7. **Making a Directory or Zip File Runnable As a Main Script**
    8. **Reading Datafiles Within a Package**
    9. **Adding Directories to sys.path**
    10. **Importing Modules Using a Name Given in a String**
    11. **Loading Modules from a Remote Machine Using Import Hooks**
    12. **Patching Modules on Import**
    13. **Installing Packages Just for Yourself**
    14. **Creating a New Python Environment**
    15. **Distributing Packages**
11. **Chapter 11 Network and Web Programming** 
    1. **Interacting with HTTP Services As a Client**
    2. **Creating a TCP Server**
    3. **Creating a UDP Server**
    4. **Generating a Range of IP Addresses from a CIDR Address**
    5. **Creating a Simple REST-Based Interface**
    6. **Implementing a Simple Remote Procedure Call with XML-RPC**
    7. **Communicating Simply Between Interpreters**
    8. **Implementing Remote Procedure Calls**
    9. **Authenticating Clients Simply**
    10. **Adding SSL to Network Services**
    11. **Passing a Socket File Descriptor Between Processes**
    12. **Understanding Event-Driven I/O**
    13. **Sending and Receiving Large Arrays**
12. **Chapter 12 Concurrency** 
    1. **Starting and Stopping Threads**
    2. **Determining If a Thread Has Started**
    3. **Communicating Between Threads**
    4. **Locking Critical Sections**
    5. **Locking with Deadlock Avoidance**
    6. **Storing Thread-Specific State**
    7. **Creating a Thread Pool**
    8. **Performing Simple Parallel Programming**
    9. **Dealing with the GIL (and How to Stop Worrying About It)**
    10. **Defining an Actor Task**
    11. **Implementing Publish/Subscribe Messaging**
    12. **Using Generators As an Alternative to Threads**
    13. **Polling Multiple Thread Queues**
    14. **Launching a Daemon Process on Unix**
13. **Chapter 13 Utility Scripting and System Administration** 
    1. **Accepting Script Input via Redirection, Pipes, or Input Files**
    2. **Terminating a Program with an Error Message**
    3. **Parsing Command-Line Options**
    4. **Prompting for a Password at Runtime**
    5. **Getting the Terminal Size**
    6. **Executing an External Command and Getting Its Output**
    7. **Copying or Moving Files and Directories**
    8. **Creating and Unpacking Archives**
    9. **Finding Files by Name**
    10. **Reading Configuration Files**
    11. **Adding Logging to Simple Scripts**
    12. **Adding Logging to Libraries**
    13. **Making a Stopwatch Timer**
    14. **Putting Limits on Memory and CPU Usage**
    15. **Launching a Web Browser**
14. **Chapter 14 Testing, Debugging, and Exceptions** 
    1. **Testing Output Sent to stdout**
    2. **Patching Objects in Unit Tests**
    3. **Testing for Exceptional Conditions in Unit Tests**
    4. **Logging Test Output to a File**
    5. **Skipping or Anticipating Test Failures**
    6. **Handling Multiple Exceptions**
    7. **Catching All Exceptions**
    8. **Creating Custom Exceptions**
    9. **Raising an Exception in Response to Another Exception**
    10. **Reraising the Last Exception**
    11. **Issuing Warning Messages**
    12. **Debugging Basic Program Crashes**
    13. **Profiling and Timing Your Program**
    14. **Making Your Programs Run Faster**
15. **Chapter 15 C Extensions** 
    1. **Accessing C Code Using ctypes**
    2. **Writing a Simple C Extension Module**
    3. **Writing an Extension Function That Operates on Arrays**
    4. **Managing Opaque Pointers in C Extension Modules**
    5. **Defining and Exporting C APIs from Extension Modules**
    6. **Calling Python from C**
    7. **Releasing the GIL in C Extensions**
    8. **Mixing Threads from C and Python**
    9. **Wrapping C Code with Swig**
    10. **Wrapping Existing C Code with Cython**
    11. **Using Cython to Write High-Performance Array Operations**
    12. **Turning a Function Pointer into a Callable**
    13. **Passing NULL-Terminated Strings to C Libraries**
    14. **Passing Unicode Strings to C Libraries**
    15. **Converting C Strings to Python**
    16. **Working with C Strings of Dubious Encoding**
    17. **Passing Filenames to C Extensions**
    18. **Passing Open Files to C Extensions**
    19. **Reading File-Like Objects from C**
    20. **Consuming an Iterable from C**

Diagnosing Segmentation Faults