

Ruby

1. Introduction to Computer Science

1.1 Introduction

1.2 Application Development

Step 1: Understand the Problem

Step 2: Write Out the Solution in Plain Language

Step 3: Translate the Language into Code

Step 4: Test the Code in the Computer

1.3 Algorithms

1.3.1 Algorithm Efficiency

1.4 Summary

1.4.1 Key Concepts

1.4.2 Key Definitions

1.5 Exercises

2. How Does the Computer Really Work?

2.1 Introduction

2.2 Basic Nomenclature and Components of a Computer System

2.3 Scales of Magnitude

2.4 Instruction Execution—Speed and Timing Scales

2.5 Bit Strings and Their Meaning

2.6 The Interpreter Process and Ruby

2.7 Summary

2.7.1 Key Concepts

2.7.2 Key Definitions

2.8 Exercises

3. Core Programming Elements

3.1 Introduction

3.2 Getting Started

How to Install Ruby

How to Save Programs

3.3 What Is a Variable?

Constants: Variables That Never Change

Data Types

Integer

Float

Strings

Booleans

3.4 Basic Arithmetic Operators

3.5 Input and Output

Output Using Variables

Display User Input

Basic Programs

Step 1: Understanding the Problem

Step 2: Write Out the Problem in Plain Language

Step 3: Rewrite the Plain Language into Code

Step 4: Test the Code in the Computer

3.6 Common Programming Errors

Syntax Errors

Logic Errors

3.7 Mixing Data Types

3.8 Summary

3.8.1 Key Concepts

3.8.2 Key Definitions

3.9 Exercises

4. Conditional Structures

4.1 Introduction

4.2 Flow of Execution

Logic Flow

4.3 Conditional Control

Control Flow

4.4 If-Then-Else Statements

Testing Conditional Flow

Elsif Statements

4.5 Case Statements

4.6 Debugging

4.6.1 Alternative Styles of Debugging

4.7 Summary

4.7.1 Key Concepts

4.7.2 Key Definitions

4.8 Exercises

5. Loop Structures. . . .

5.1 Introduction

5.2 While Loops

5.3 Until Loops

5.4 For Loops and Nested Loops

For Loops

Nested Loops

5.5 Infinite Loops

5.6 Example: Finding Prime Numbers

5.7 Summary

- 5.7.1 Key Concepts
- 5.7.2 Key Definitions
- 5.8 Exercises

6. Arrays

- 6.1 Introduction
- 6.2 Array Types
 - 6.2.1 One-Dimensional Arrays
 - Example: Find the Max
 - 6.2.2 Multidimensional Arrays
 - Example: Find the Max–Modified
- 6.3 Hashes
 - Example: Hash
 - Example: Accessing a Hash
 - Example: Find the Max–Hash
- 6.4 Summary
 - 6.4.1 Key Concepts
 - 6.4.2 Key Definitions
- 6.5 Exercises

7. Sorting and Searching.

- 7.1 Introduction
 - 7.1.1 Selection Sort
 - 7.1.2 Insertion Sort
 - 7.1.3 Bubble Sort
 - 7.1.4 Radix Sort
- 7.2 Complexity Analysis
- 7.3 Searching

7.3.1 Linear Search

7.3.2 Binary Search

7.4 Summary

7.4.1 Key Concepts

7.4.2 Key Definitions

7.5 Exercises

8. Using Objects

8.1 Introduction

8.2 Objects and Built-in Objects

8.2.1 Objects

8.2.2 Built-in Objects

8.2.3 Parameter Passing

8.3 Summary

8.3.1 Key Concepts

8.3.2 Key Definitions

8.4 Exercises

9. Defining Classes and Creating Objects

9.1 Introduction

9.2 Instantiating Objects from Classes

9.3 Data and Methods

9.3.1 Grouping Data and Methods

9.3.2 Implementing Methods

9.4 Summary

9.4.1 Key Concepts

9.4.2 Key Definitions

9.5 Exercises

10. Object Inheritance

10.1 Introduction

10.2 Inheritance

10.3 Basic Method Overriding

10.4 Accessing the Superclass

10.5 Applications

10.5.1 Person Database

10.5.2 Grocery Store

10.5.3 Video Games

10.6 Summary

10.6.1 Key Concepts

10.6.2 Key Definitions

10.7 Exercises

11. File Input/Output.

11.1 Introduction

11.2 File Access: Reading and Writing

11.2.1 File Reader Class

11.2.2 FileWriter Class

11.2.3 File Reader/Writer Example

11.3 Summary

11.3.1 Key Concepts

11.3.2 Key Definitions

11.4 Exercises

12. Putting It All Together: Tic-Tac-Toe

12.1 Introduction

12.2 Programming Approach

12.3 Tic-Tac-Toe

12.4 Tic-Tac-Toe Revised

12.5 Summary

12.6 Exercises