TypeScript Algorithms

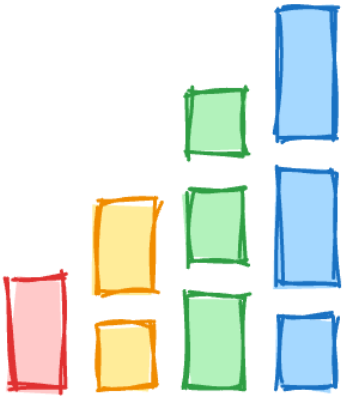
Contents

1 Introduction **3**

2 Algorithms and Data Structures **4**
- 2.1 Algorithm Analysis . . . 5
- 2.2 Bits . . . 5
  - 2.2.1 Bit Parity . . . 5
  - 2.2.2 Bit Shift Operator . . . 6
- 2.3 Recursion . . . 6
  - 2.3.1 Fibonacci Sequence . . . 6

3 Domain Specific **7**
- 3.1 Language . . . 7
  - 3.1.1 This . . . 7
  - 3.1.2 Event Loop . . . 7
  - 3.1.3 Asynchronous Programming . . . 7
  - 3.1.4 Runtime Environments . . . 7

4 Appendix **8**
- 4.1 Resources . . . 8
- 4.2 Contributing . . . 8
  - 4.2.1 Getting Started . . . 8
  - 4.2.2 Commands . . . 9



TypeScript

ALGORITHMS

Chapter 1

Introduction

In-progress book about algorithms and data structures in TypeScript.

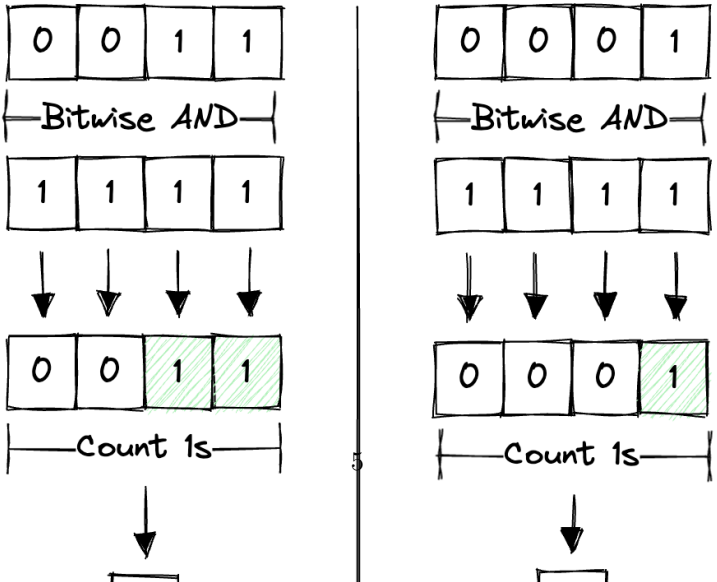
Chapter 2

Algorithms and Data Structures

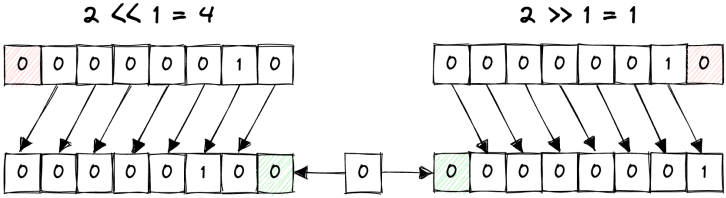
2.1 Algorithm Analysis

2.2 Bits

2.2.1 Bit Parity

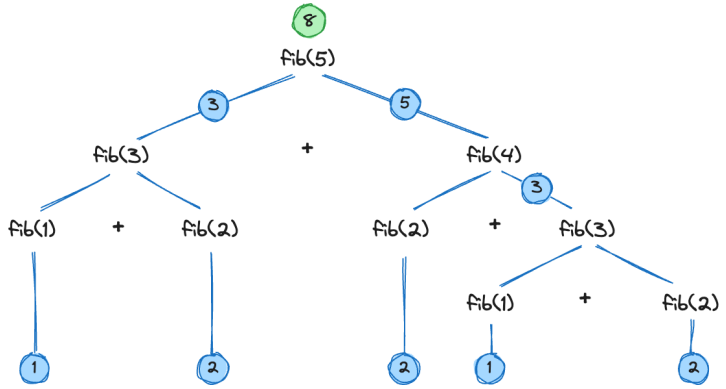


2.2.2 Bit Shift Operator



2.3 Recursion

2.3.1 Fibonacci Sequence



$$fib_n = fib_{n-2} + fib_{n-1}$$

Chapter 3

Domain Specific

3.1 Language

3.1.1 This

3.1.2 Event Loop

3.1.3 Asynchronous Programming

3.1.3.1 Promises

3.1.3.2 Async/Await

3.1.4 Runtime Environments

3.1.4.1 Browser

3.1.4.2 Server

Chapter 4

Appendix

4.1 Resources

- LeetCode
- Project Euler
- The Algorithm Design Manual
- Elements of Programming Interviews

4.2 Contributing

4.2.1 Getting Started

4.2.1.1 Run Problem Tests

- Install Node Version Manager
- `yarn setup`
- `yarn test --all`

4.2.1.2 Compile Book

- Install Pandoc (Homebrew)
- Install BasicTeX (Homebrew)

- Install fswatch (Homebrew)
- `make book`
- `open book/output/index.pdf`

4.2.2 Commands

Command	Description
`yarn setup`	Setup local development environment
`yarn test`	Run tests
`yarn gen:leetcode`	Generate new LeetCode problem
`yarn gen:project-euler`	Generate new Project Euler problem
`make book`	Compile book to various formats
`make watch`	Recompile book automatically when source files change