

TypeScript Algorithms

Contents

| | | |
|----------|---------------------------------------|----------|
| 1 | Introduction | 2 |
| 2 | Algorithms and Data Structures | 3 |
| 2.1 | Algorithm Analysis | 3 |
| 2.2 | Bits | 3 |
| 2.2.1 | Bit Parity | 3 |
| 2.2.2 | Bit Shift Operator | 3 |
| 2.3 | Recursion | 3 |
| 2.3.1 | Fibonacci Sequence | 3 |
| 3 | Domain Specific | 4 |
| 3.1 | Language | 4 |
| 3.1.1 | This | 4 |
| 3.1.2 | Event Loop | 4 |
| 3.1.3 | Asynchronous Programming | 4 |
| 3.1.4 | Runtime Environments | 4 |
| 4 | Appendix | 5 |
| 4.1 | Resources | 5 |
| 4.2 | Contributing | 5 |
| 4.2.1 | Getting Started | 5 |
| 4.2.2 | Commands | 6 |

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