



TypeScript

ALGORITHMS

Version 0.0.1

Contents

1	Introduction	2
2	Algorithm Analysis	3
3	Data Structures and Algorithms	4
3.1	Bits	5
3.1.1	Overview	5
3.1.2	Bit Parity	5
3.1.3	Bit Shift Operator	6
3.2	Stacks and Queues	6
3.2.1	Overview	6
3.2.2	Fixed Stack	6
4	Problem Solving Methods	7
4.1	Recursion	7
4.1.1	Overview	7
4.1.2	Fibonacci Sequence	7
5	Domain Specific	9
5.1	Language	9
5.1.1	This	9
5.1.2	Event Loop	9
5.1.3	Asynchronous Programming	9
5.1.4	Runtime Environments	9
6	Appendix	10
6.1	Resources	10

Chapter 1

Introduction

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

Chapter 2

Algorithm Analysis

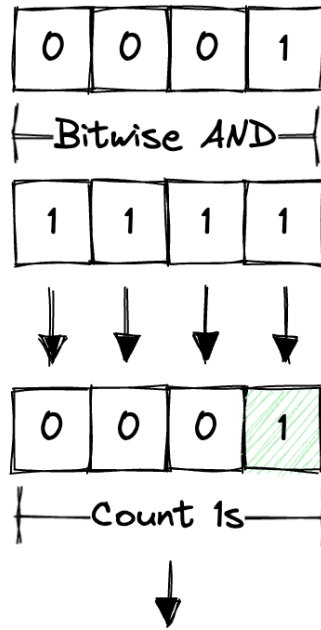
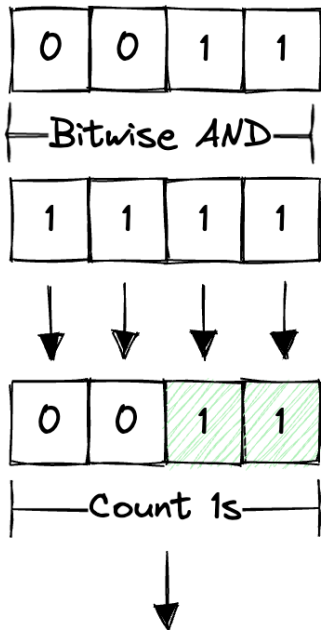
Chapter 3

Data Structures and Algorithms

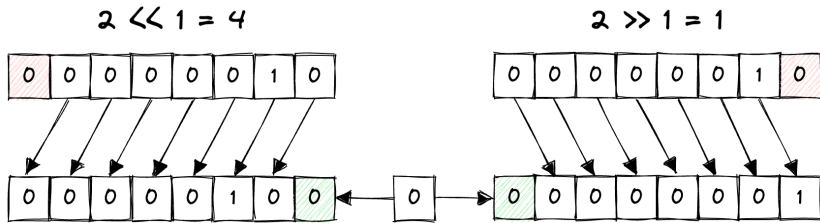
3.1 Bits

3.1.1 Overview

3.1.2 Bit Parity



3.1.3 Bit Shift Operator



3.2 Stacks and Queues

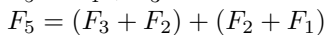
3.2.1 Overview

3.2.2 Fixed Stack

Problem Solving Methods

4.1.2 Fibonacci Sequence

$$F_n = F_{n-1} + F_{n-2} \quad for \ n > 1$$



$$F_5 = ((F_2 + F_1) + (F_1 + F_0)) + ((F_1 + F_0) + F_1)$$

$$F_5 = (((F_1 + F_0) + F_1) + (F_1 + F_0)) + ((F_1 + F_0) + F_1)$$

$$F_5 = (((1 + 0) + 1) + (1 + 0)) + ((1 + 0) + 1)$$

$$F_5 = 5$$

```
export function fib(n: number): number {
  if (n == 0 || n == 1) {
    return n
  }
  return fib(n - 1) + fib(n - 2)
}
```

Chapter 5

Domain Specific

5.1 Language

5.1.1 This

5.1.2 Event Loop

5.1.3 Asynchronous Programming

5.1.3.1 Promises

5.1.3.2 Async/Await

5.1.4 Runtime Environments

5.1.4.1 Browser

5.1.4.2 Server

Chapter 6

Appendix

6.1 Resources

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