Exercise/Problem Guide for *Nature of Computation* by Moore and Mertens

Muthu Chidambaram

Last Updated: June 21, 2019

Contents

About

> *"Computer science is no more about computers than astronomy is about telescopes."* - Edsger Dijkstra (Maybe)

These notes contain my solutions to some exercises and problems from the book *Nature of Computation* by Christopher Moore and Stephan Mertens.

1 Insights and Algorithms

1.1 Exercises

1.1.1 Exercise 3.1

Assume $f(n) = 2^n - 1$. Then $f(n+1) = 2^{n+1} - 2 + 1 = 2^{n+1} - 1$.

1.1.2 Exercise 3.2

We have that

$$\begin{aligned}(QQ^*)_{ab} &= \frac{1}{n}\sum_{k=0}^{n-1} w_n^{ak} w_n^{\bar{k}b} \\ &= \frac{1}{n}\sum_{k=0}^{n-1} w_n^{k(i-j)}\end{aligned}$$

If $i = j$, then $(QQ^*)_{ab} = 1$. Otherwise, since $w_n^{(i-j)}$ is a root of unity,

$$(QQ^*)_{ab} = w_n^{i-j}(QQ^*)_{ab} \implies (1 - w_n^{i-j})(QQ^*)_{ab} = 0 \implies (QQ^*)_{ab} = 0$$

And we have that $Q^* = Q^{-1}$ as desired.

2 Appendix

2.1 Exercises

2.1.1 Exercise A.1

We can choose $n_3 = max(n_1, n_2)$ such that $f_1(n)+f_2(n) \leq (C_1+C_2)g$ whenever $n > n_3$.

2.1.2 Exercise A.2

Similar to the first exercise, but now we have C_1C_2h instead.

2.1.3 Exercise A.3

Logarithms of different bases differ by a constant.

2.1.4 Exercise A.4

The argument treats k as a constant, when in reality $k = O(n)$. The answer should be $O(n^3)$, which can be checked by looking at the closed form of the sum.

2.1.5 Exercise A.5

$2^{O(n)}$ and $O(2^n)$ are not the same, since $\limsup_{n\to\infty} \frac{2^{C_1 n}}{C_2 2^n}$ only converges to a nonzero constant when $C_1 = 1$.

2.1.6 Exercise A.6

$n^k \neq \Theta(2^n)$, $e^{-n} \neq \Theta(n^{-c})$, and $n! \neq \Theta(n^n)$, as all limits go to 0.

2.1.7 Exercise A.7

Take $f = n$ and $g = 2n$.

2.1.8 Exercise A.8

1. We can pull out the exponents as constants, so $f = \Theta(g)$.
2. $3 < 2^2$ so $f = o(g)$.
3. $n = \omega(\log^2 n)$ so $f = \omega(g)$.
4. $2^{\log n} \to \infty$ so $f = \omega(g)$.

2.1.9 Exercise A.9

One example is $n^{\log n}$.