CSE 102 - Midterm Study Guide

Zack Traczyk

Vaggos, Winter 2024

Contents

1 Introductory Material Review **2**
1.1 Asymptotic Bounds 2
1.2 Inductive Proofs 2
1.3 Practice Problems 3
1.3.1 HW1 - Ex.1 Part 2 3

2 Solving Recurrence Relations **3**
2.1 Master Theorem 3
2.2 Unpacking Tree / Algebraic Pattern 3
2.3 Substitution 3
2.4 Guess and Verify 3
2.5 Practice Problems 3
2.5.1 HW3 - Ex.4 3

3 Algorithms **7**
3.1 Binary Search 7
3.2 Sorting 7
3.2.1 Lower Bounds 7
3.3 Merge Sort 7
3.4 Number of leaves / depth as proof for lower asymptotic bounds 7
3.5 Quick Select 7
3.6 Dynamic Programming 7
3.6.1 Fibonacci 7
3.6.2 Binomial Coefficients 7
3.6.3 Maximize independent set 7

1 Introductory Material Review

1.1 Asymptotic Bounds

Definition 1 (Big-O)**.** $f(n) = O(g(n))$ if there exists a positive constant c and an integer n_0 such that $f(n) \leq c \cdot g(n)$ for all $n \geq n_0$.

Definition 2 (Big-Ω)**.** $f(n) = \Omega(g(n))$ if there exists a positive constant c and an integer n_0 such that $c \cdot g(n) \leq f(n)$ for all $n \geq n_0$.

Definition 3 (Big-Θ)**.** $f(n) = \Theta(g(n))$ if there exists positive constants c_1, c_2, and an integer n_0 such that $c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n)$ for all $n \geq n_0$.

1.2 Inductive Proofs

Definition 4 (Inductive Proof)**.** Simples rules of induction taken from CSE 16 with Prof. Tracy Larrabee.

1. Write down the Left Hand Side of $P(k+1)$.
2. Rewrite $P(k+1)$ to include Left Hand Side $P(k)$.
3. Replace Left Hand Side of $P(k)$ with Right Hand Side of $P(k)$.
4. Rewrite so Right Hand Side of $P(k)$ becomes Right Hand Side of $P(k+1)$.

Examples:

Proof. For all $n \in \mathbb{Z}^+$ the number $n^2 + n$ is even
Base Case (n = 1):

$$n^2 + n = 1 + 1 = 2, \text{ which is even}$$

Inductive Hypothesis:

$$\text{Assume } n^2 + n \text{ is even, prove } (n+1)^2 + (n+1) \text{ is even}$$

$$\begin{aligned}
(n+1)^2 + (n+1) &= n^2 + 2n + 1 + n + 1 && (1)\\
&= n^2 + n + 2n + 2 && (2)\\
2p = n^2 + n&, \text{ 2p is the definition of even} && (3)\\
&= 2p + 2n + 2 && (4)\\
&= 2(p + n + 1), \text{ which is even} && (5)
\end{aligned}$$

□

1.3 Practice Problems

1.3.1 HW1 - Ex.1 Part 2

Problem 1.1. Prove that $T(n) = 2T(n-1) + 1$ is $T(n) = 2^n - 1$.

2 Solving Recurrence Relations

2.1 Master Theorem

2.2 Unpacking Tree / Algebraic Pattern

2.3 Substitution

2.4 Guess and Verify

2.5 Practice Problems

2.5.1 HW3 - Ex.4

Like in many previous exercises and homeworks, find tight asymptotic bounds (big-Theta) for $T(n)$ in each of the cases.

Problem 2.1. $T(n) = 2T(n/4) + n^2\sqrt{n}$

Problem 2.2. $T(n) = T(n-1) + \frac{1}{n}$

Problem 2.3. $T(n) = 1600T(n/4) + n!$ (hint: answering this shouldn't require too many, if any, difficult calculations)

Problem 2.4. $T(n) = 6T(n/3) + n^4/\log^{25} n$ (hint: answering this shouldn't require too many, if any, difficult calculations)

Problem 2.5. $T(n) = \sqrt{n}T(\sqrt{n}) + n$ (hint: when everything fails, you guess and check)

Problem 2.6. $T(n) = T(n/2) + n(5 - \cos^2 n \sin^{20} n)$ (hint: answering this shouldn't require too many, if any, difficult calculations, just think the most basic trigonometric inequality)

Problem 2.7. $T(n) = \alpha T(n/4) + n^2$ (hint: your answer should depend on the α parameter)

Problem 2.8. $T(n) = 5T(n/5) + \frac{n}{\log_5 n}$ (hint: think of $n = 5^m$. Also the recursion $T(n) = T(n-1) + \frac{1}{n}$ above may come in handy.)

3 Algorithms

3.1 Binary Search

3.2 Sorting

3.2.1 Lower Bounds

3.3 Merge Sort

3.4 Number of leaves / depth as proof for lower asymptotic bounds

3.5 Quick Select

3.6 Dynamic Programming

3.6.1 Fibonacci

3.6.2 Binomial Coefficients

3.6.3 Maximize independent set