Homework 1

Name: Rohan Karamel
NetID: rak218
Course: Algorithms 2
Instructor: Professor Szegedy
Date: February 13, 2024

Problem 1. Book 0.1

Solution.

a $f = \Theta(g)$

b $f = O(g)$

c $f = \Theta(g)$

d $f = \Theta(g)$

e $f = \Theta(g)$

f $f = \Theta(g)$

g $f = O(g)$

h $f = \Omega(g)$

i $f = \Omega(g)$

j $f = \Omega(g)$

k $f = O(g)$

l $f = O(g)$

m $f = \Theta(g)$

n $f = \Omega(g)$

o $f = \Omega(g)$

p $f = O(g)$

Problem 2. Book 0.2

Solution.

a If c is less than 1, $g(n)$ becomes a geometric series with common ratio, c. This series is equivalent to $\frac{1-c^n}{1-c}$ which is bounded between 0 and $\frac{1}{1-c}$. Therefore, it is $\Theta(1)$.

b If c is equal to 1, then all terms of the series will also be 1. Because there are n terms, the series converges to n. Therefore, it is $\Theta(n)$.

c We only care about the dominant term in the series, we will drop terms with lower power and focus on the c^n. Therefore, it is $\Theta(c^n)$

Problem 3. Book 1.11: Is $4^{1536} - 9^{4824}$ divisible by 35?

Solution. We can show it is divisible by taking each term modulo 35.

$$4^{6k} \equiv 1 \pmod{35} \forall k \in \mathbb{Z}$$

$$4^{1536} \equiv 4^{6 \cdot 256} \equiv 1 \pmod{35}$$

Similarly,

$$9^{6k} \equiv 1 \pmod{35}$$

$$9^{4824} \equiv 9^{6 \cdot 804} \equiv 1 \pmod{35}$$

Therefore, we can simplify the original statement

$$4^{1536} - 9^{4824} \equiv 1 - 1 \equiv 0 \pmod{35}$$

Therefore $4^{1536} - 9^{4824}$ is divisible by 35.

Problem 4. Book 1.12: What is $2^{2^{2023}} \pmod{3}$?

Solution. Notice that

$$2 \equiv -1 \pmod{3}$$

$$2^k \equiv (-1)^k \pmod{3}$$

If k is even, we can simplify this to

$$2^k \equiv 1 \pmod{3}$$

Because 2^{2023} is even, we can set k equal to 2^{2023}

$$2^{2^{2023}} \equiv 1 \pmod{3}$$

And we are done.

Problem 5. Book 1.14: Suppose you want to compute the n^{th} Fibonacci number modulo 5. Describe the most efficient way in which you can do this.

Solution. We begin by solving the Fibonacci recurrence relation. This yields the following formula:

$$F_n = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n + \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n$$

This formula requires us to only calculate exponentiation for the n^{th} Fibonacci number. Because exponentiation can be done in logarithmic time and exponentiation being the most time-consuming action here, the algorithm would be $O(\log(n))$. Note that exponentiation only runs in logarithmic time for small n, this means that for large n, the complexity changes.

Problem 6. Grad student A has designed an algorithm whose running time is $\log(n)^{\log(n)}$. Grad student B has designed an algorithm whose running time is $\frac{n}{\log(n)}$. Which student has the better algorithm as n goes to ∞?

Solution. Grad student B has a vastly better algorithm as $n \to \infty$. B's algorithm is faster than linear time and A's is worse than exponential time.

Problem 7. Book 1.17

Solution. The iterative approach requires $y-1$ multiplications therefore, in terms of y and n, this algorithm has a running time of $n^2(y-1)$. The recursive approach would have a running time of $n^2\log(y-1)$ because there are $\log(y-1)$ multiplications. Overall, the recursive approach is more efficient as it runs in logarithmic time.

Problem 8. Book 1.20

Solution.

a 4 because $4 \cdot 20 = 80 \equiv 1 \pmod{79}$

b 21 because $21 \cdot 3 = 63 \equiv 1 \pmod{62}$

c The inverse does not exist because 21 and 91 are not coprime as they are both divisible by 7.

d 14 because $14 \cdot 5 = 70 \equiv 1 \pmod{23}$