6.1200J/18.062J *Mathematics for Computer Science* Tuesday 27th February, 2024
Massachusetts Institute of Technology, Spring 2024
Z. Abel, B. Chapman, E. Demaine revised Monday 26th February, 2024

Problem Set 4

- **Due Date:** 11:59pm on **Monday 4th March, 2024**
- **Days Covered**: 06 and 07 (including Lecture, Warm-Up, and Recitation)

Problem 1. Asymptotic Relationships [15 points]

(a) [7 pts] For each pair of functions $f : \mathbb{N}^+ \to \mathbb{R}^+$ and $g : \mathbb{N}^+ \to \mathbb{R}^+$ in the table below, list in the third column which of the following is the strongest or most specific asymptotic relationship that holds between f and g, from among these options:

$$f \sim g, \qquad f \in \Theta(g), \qquad f \in O(g), \qquad f \in o(g), \qquad f \in \omega(g), \qquad f \in \Omega(g), \qquad \text{None.}$$

You do not need to explain your work in this part. (Assume that log means $\log_2$ if no base is specified.)

$f(n)$	$g(n)$	Relationship?
6^n	n^{1200}	
$(\log n)^6$	$\sqrt[1200]{n}$	
$\log_6(n^{1200})$	$(\log_{1200} n)^6$	
$\log(n!)$	$n \log n$	
$2^{\lceil \log_2 n \rceil}$	$3^{\lceil \log_3 n \rceil}$	
$2^{\lceil \log_2 n \rceil} - n + 1$	$3^{\lceil \log_3 n \rceil} - n + 1$	
n	$n^{\cos \log n}$	

(b) [8 pts] Answer the same question for the pairs below, and **justify** your reasoning.

$f(n)$	$g(n)$	Relationship?
$2^{((\log n)^2)}$	$2^{(\lceil \log n \rceil^2)}$	
4^n	$(2n)!/(n!)^2$	

Problem 2. OEIS [10 points]

(a) [2 pts] Consider the sequence that begins with $x_0 = 0, x_1 = 1, x_2 = 2$, and continues with $x_n = 5x_{n-2} + 2x_{n-3}$ for $n \geq 3$. What are the values of x_3, x_4, x_5, x_6?

(b) [1 pts] Let's practice one of the most useful techniques when investigating a sequence or recurrence: checking whether someone else has already done it for you! The OEIS (Online Encyclopedia of Integer Sequences) is an invaluable tool that houses more than 300,000 noteworthy integer sequences together with descriptions, formulas, external references, relationships to other OEIS entries, etc. It continues to grow as users contribute more sequences and provide more information on existing entries.

Use `https://oeis.org/` to look up an entry that might possibly match our sequence from part (a): type the values we've computed into their search bar, separated by commas. If you're not sure you've found the right one, you can gain confidence by computing more terms using our recurrence and seeing if they continue to match.

(c) [1 pts] Browse your chosen entry (scroll down to the "Formula" section) to find a possible *closed form* function of n that equals x_n. Here, *closed form* means that the function can be computed directly from n without relying on other terms in the sequence or iterative processes such as $\sum$, $\prod$, or "dot-dot-dot"s. Basic arithmetic functions such as exponents are fine.

Note: lines that include the acronyms "g.f." or "e.g.f.", which stand for *generating function* and *exponential generating function*, can be safely ignored for this problem.

(d) [6 pts] Don't trust everything you read online! Use induction to carefully prove the formula you found in the previous part.

Problem 3. Bounded Asymptotics [15 points]

In this problem, we investigate the asymptotic relationship of functions under some conditions. You can assume that any function mentioned in the problem maps $\mathbb{N}$ to $\mathbb{R}^+$. You might find the following definitions useful:

1. A function f is bounded from below (respectively above) if there exists a real number B such that $f(n) \geq B$ (respectively $f(n) \leq B$) for all $n \in \mathbb{N}$.
2. A function f is bounded away from zero if there exists a real number $B > 0$ such that $f(n) \geq B$ for all $n \in \mathbb{N}$.
3. A funciton f is strictly increasing (respectively decreasing) if for any $a, b \in \mathbb{N}$ we have that $a > b \Rightarrow f(a) > f(b)$ (respectively $a > b \Rightarrow f(a) < f(b)$).

(a) [5 pts] Let f be a function that is bounded from above and g be a function that is bounded away from zero. Carefully prove that $f \in O(g)$.

Hint: Careful! The limit $\lim\limits_{n\to\infty} \frac{f(n)}{g(n)}$ might not exist, so your argument shouldn't use limits.

(b) [5 pts] Let f be a function that is bounded from above and g be a function that is bounded from below. Is it necessarily true that $f \in O(g)$?

(c) [5 pts] Let f be a function that is strictly decreasing and g be a function that is strictly increasing. Is it necessarily true that $f \in O(g)$? Is it necessarily true that $f \in o(g)$?

Problem 4. Karatsuba's Algorithm [10 points]

As we'll see soon, cryptography often requires arithmetic with large integers, possibly with hundreds or thousands of digits each! When adding two integers A and B each with n digits, the straightforward algorithm of adding digits from right-to-left, carrying into the next column if needed, requires at most $2n$ single-digit additions. But what about multiplication? The straightforward multiplication algorithm involves multiplying each digit of A with each digit of B, which is already $\Theta(n^2)$ operations! (Not to mention the $\Theta(n^2)$ single-digit additions that follow.) Karatsuba's Algorithm is a clever divide-and-conquer algorithm with (hopefully) faster asymptotic runtime.

Karatsuba's Algorithm is (optionally) described below. It can multiply two n-digit numbers using at most $T(n)$ single-digit arithmetic operations, where $T(n)$ is defined by the following recurrence:

$$T(0) = 0, \qquad T(1) = 1, \qquad T(n) = 3T(\lceil n/2 \rceil) + 10n \quad \text{for } n \geq 2.$$

(a) [1 pts] Compute $T(2)$, $T(3)$, and $T(4)$.

(b) [4 pts] **Use the Plug and Chug method** to find an exact (not just asymptotic!) closed-form expression for $T(n)$, assuming n is an exact power of 2. You do not need to prove your formula, but you should check it against $T(1)$, $T(2)$, and $T(4)$.

Note: We'd like to see you use the Plug and Chug method specifically; solutions that arrive at a closed form expression using other techniques (or without explanation) will only receive partial credit.

(c) [4 pts] Karatsuba's algorithm works whether or not n is a power of 2, and the same recurrence applies, so let's find an approximate solution for $T(n)$ in this more general case: carefully use Master Theorem to prove an asymptotic bound of the form $T(n) \in \Theta(g(n))$, for some closed-form function $g(n)$.

(d) [1 pts] Is Karatsuba an asymptotic improvement over the straightforward $\Theta(n^2)$ algorithm? Why or why not?

Optional Note: Here's a rough description of Karatsuba's Algorithm, in case you're curious. If A and B are n-digit numbers, we can split each list of digits in half by writing $A = A_0X + A_1$ and $B = B_0X + B_1$, where $X = 10^{\lceil n/2 \rceil}$. (For example, when $A = 314159265$ and $B = 271828182$, then we're simply writing $A = 3141 \cdot X + 59265$ and $B = 2718 \cdot X + 28182$, where $X = 10^5$.) We could compute $A \cdot B$ using the formula

$$A \cdot B = \underbrace{[A_0 \cdot B_0]}_{P} X^2 + \underbrace{([A_0 \cdot B_1] + [A_1 \cdot B_0])}_{Q} X + \underbrace{[A_1 \cdot B_1]}_{R}.$$

This requires (some additions, which are cheap, and) 4 multiplications of $\lceil n/2 \rceil$-digit numbers shown in square brackets, which are computed recursively. This leads to a recurrence of the form $T(n) = 4T(\lceil n/2 \rceil) + O(n)$, which has solution $T(n) = \Theta(n^2)$. Karatsuba cleverly observed that P, Q, R can be computed with just three multiplications, at the expense of a few more additions and subtractions (which are still cheap):

$$\begin{aligned} P &= [A_0 \cdot B_0] \\ R &= [A_1 \cdot B_1] \\ Q &= P + R - [(A_0 - A_1) \cdot (B_0 - B_1)]. \end{aligned}$$

This reduces the 4 to a 3 in the recurrence, while the $O(n)$ term remains $O(n)$ (though with a larger constant factor), which directly leads to the asymptotic speedup.