# CS 170 Homework 1

Due **Friday 9/6/204, at 10:00 pm (grace period until 11:59pm)**

## 1 Study Group

List the names and SIDs of the members in your study group. If you have no collaborators, explicitly write "none".

## 2 Course Policies

(a) What dates and times are the exams for CS170 this semester? Are there planned alternate exams?

**Solution:**

(a) **Midterm 1**: Wednesday, 10/2/2024, 7:00 PM - 9:00 PM

(b) **Midterm 2**: Tuesday, 11/5/2024, 7:00 PM - 9:00 PM

(c) **Final**: Tuesday, 12/17/2024, 8:00 AM - 11:00 AM

We do not plan on offering alternate exams.

(b) Homework is due Fridays at 10:00pm, with a late deadline at 11:59pm. At what time do we recommend you have your homework finished?

**Solution:** 10:00pm

(c) We provide 2 homework drops for cases of emergency or technical issues that may arise due to homework submission. If you miss the Gradescope late deadline (even by a few minutes) and need to submit the homework, what should you do?

**Solution:** The 2 homework drops are provided in case you have last minute issues and miss the Gradescope deadline. Homework extensions are not granted because solutions need to be released soon after the deadline, and so you do nothing.

However, we do give exceptions for DSP-related reasons and/or severe circumstances that affect multiple weeks at a time. If you believe you qualify for those, please contact cs170@.

(d) What is the primary source of communication for CS170 to reach students? We will send out all important deadlines through this medium, and you are responsible for checking your emails and reading each announcement fully.

**Solution:** The primary source of communication is Edstem.

(e) Please read all of the following:

(i) **Syllabus and Policies:** `https://cs170.org/policies/`

(ii) **Homework Guidelines:** `https://cs170.org/resources/homework-guidelines/`

(iii) **Regrade Etiquette:** `https://cs170.org/resources/regrade-etiquette/`

(iv) **Forum Etiquette:** `https://cs170.org/resources/ed-etiquette/`

Once you have read them, copy and sign the following sentence on your homework submission.

"I have read and understood the course syllabus and policies."

**Solution:** I have read and understood the course syllabus and policies. -Alan Turing

# 3 Understanding Academic Integrity

Before you answer any of the following questions, make sure you have read over the syllabus and course policies (`https://cs170.org/policies/`) carefully. For each statement below, write *OK* if it is allowed by the course policies and *Not OK* otherwise.

(a) You ask a friend who took CS 170 previously for their homework solutions, some of which overlap with this semester's problem sets. You look at their solutions, then later write them down in your own words.

**Solution:** Not OK.

(b) You had 5 midterms on the same day and are behind on your homework. You decide to ask your classmate, who's already done the homework, for help. They tell you how to do the first three problems.

**Solution:** Not OK.

(c) You're a serial procrastinator and started working on the homework at 9:00 PM on Friday, and out of desperation searched up a homework problem online and find the exact solution. You then write it in your words and cite the source.

**Solution:** Not OK. As a general rule of thumb, you should never be in possession of any exact homework solutions other than your own.

(d) You were looking up Dijkstra's on the internet, and inadvertently run into a website with a problem very similar to one on your homework. You read it, including the solution, and then you close the website, write up your solution, and cite the website URL in your homework writeup.

**Solution:** OK. Given that you'd inadvertently found a resource online, clearly cite it and make sure you write your answer from scratch.

# 4 Log Identities

The following subparts will cover several math identities, tricks, and techniques that will be useful throughout the rest of this course.

Simplify the following expressions into a single logarithm (i.e. in the form $\log_a b$):

(a) $\frac{\ln x}{\ln y}$

**Solution:** $\log_y x$

(b) $\ln x + \ln y$

**Solution:** $\ln xy$

(c) $\ln x - \ln y$

**Solution:** $\ln(x/y)$

(d) $170 \ln x$

**Solution:** $\ln(x^{170})$

# 5 Asymptotics Practice

For each pair of functions $f$ and $g$, specify whether $f = O(g)$, $g = O(f)$, or both. No justification needed.

(a) $f(n) = n^2 + 5n$, $g(n) = 1000(n+1)^2$.

**Solution:** Both $f = O(g)$ and $g = O(f)$.

(b) $f(n) = 5n^3$, $g(n) = n^3 + (\log n)^{10}$.

**Solution:** Both $f = O(g)$ and $g = O(f)$.

Polylogarithmic functions grow more slowly than polynomial functions, so the dominant term in $g$ is $n^3$. This term has the same degree as the dominant term in $f$, so $f = O(g)$.

(c) $f(n) = n^{100}$, $g(n) = (1.01)^n$.

**Solution:** $f = O(g)$.

(d) $f(n) = (\log n)^{10}$, $g(n) = n^{0.1}$.

**Solution:** $f = O(g)$. Again, polylogarithmic functions grow more slowly than polynomial functions, even if the polynomial term has a lower degree.

(e) $f(n) = n \cdot 2^n$, $g(n) = 3^n$.

**Solution:** $f = O(g)$. We can use L'Hospital's rule and a limit proof to show that

$$\lim_{n\to\infty} \frac{n \cdot 2^n}{3^n} = \lim_{n\to\infty} \frac{n}{(3/2)^n} = \lim_{n\to\infty} \frac{1}{(3/2)^n \log(3/2)} = 0$$

However, note that $g \neq O(f)$. Suppose for the sake of contradiction that there exists some constant $c$ such that $g(n) \leq c \cdot f(n)$ for all $n$.

However, by definition of limit, for every $c > 0$, there exists some $N$ such that for all $n > N$, $\frac{f}{g}(n) < c$. However, this means that for some $n$, $f(n) < c \cdot g(n)$ which contradicts the notion that $g = O(f)$.

(f) Consider the factorial function: $n! = 1 \cdot 2 \cdot \ldots \cdot n$.

$f(n) = n!$, $g(n) = n^n$.

**Solution:** $f = O(g)$.

$f(n) = \prod_{i=1}^{n} i \leq \prod_{i=1}^{n} n = n^n = g(n)$ for all $n$.

However, $g(n) \neq O(f(n))$.

(g) $f(n) = 1 + b + b^2 + \ldots + b^n$, $g(n) = b^n$ for arbitrary constant $b > 0$.

Does your answer change depending on the value of $b$? If so, specify the range of $b$ for which each statement holds.

**Solution:** $g = O(f)$ for all $b > 0$, as $g(n) \leq f(n)$ for all $n$.

Additionally, if $b > 1$, then $f = O(g)$ as well: $f(n)$ is a geometric series which has a closed form of $\frac{b \cdot (b^n - 1)}{b-1}$, which is $O(b^n)$ because we can ignore the constant factor of $\frac{b}{b-1}$.

If $b = 1$, then $f(n) = n$ and $g(n) = 1$, so $g \neq O(f)$.

If $b < 1$, $f \neq O(g)$ because for any constant $c > 0$, we can find an $n$ large enough that $f(n) > 1$ but $1 > c \cdot g(n)$.

## 6 Recurrence Relations

For each part, find the asymptotic order of growth of $T$; that is, find a function $g$ such that $T(n) = \Theta(g(n))$. Show your reasoning and **do not directly apply the Master Theorem; doing so will yield 0 credit**.

In all subparts, you may ignore any issues arising from whether a number is an integer.

(a) $T(n) = 2T(n/3) + 5n$

**Solution:** Expanding out $T(n)$,

$$\begin{aligned}
T(n) &= 5n + 2 \cdot \underbrace{(5n/3 + 2 \cdot T(n/9))}_{T(n/3)} && (1)\\
&= 5n + 5n \cdot 2/3 + 5n \cdot (2/3)^2 + 5n \cdot (2/3)^3 \cdots && (2)\\
&= \sum_{k=0}^{\lfloor \log_3 n \rfloor} (2/3)^k \cdot 5n && (3)\\
&\leq \sum_{k=0}^{\infty} (2/3)^k \cdot 5n && (4)\\
&= \frac{1}{1 - 2/3} \cdot 5n && (5)\\
&= \Theta(1) \cdot 5n && (6)\\
&= \Theta(n) && (7)
\end{aligned}$$

Notice that we drop a constant fraction (1/3 in this case) of the input at each level. In other words, there are 2 recursive calls on sub-problems of size $n/3$ each, so the total input size on each subsequent level is $2n/3$, where $n/3$ of the input is dropped.

Hence, $T(n)$ will turn into a geometric series. We only need the fact that the summation in (3) converges, and not its actual value, since it is a constant. So, the amount of work will be the same as the work in the first level $\Theta(n)$.

(b) An algorithm $\mathcal{A}$ takes $\Theta(n^2)$ time to partition the input into 5 sub-problems of size $n/5$ each and then recursively runs itself on 3 of those subproblems. Describe the recurrence relation for the run-time $T(n)$ of $\mathcal{A}$ and find its asymptotic order of growth.

**Solution:**

$$T(n) = 3T(n/5) + \Theta(n^2)$$

Same idea as (a), we drop a constant fraction of the input size/amount of work done in the first level, so the answer is $\Theta(n^2)$.

(c) $T(n) = T(3n/5) + T(4n/5)$ (We have $T(1) = 1$)

*Hint: first, compute a reasonable upper and lower bound for $T(n)$. Then, try to guess a $T(n)$ of the form $an^b$ and then use induction to argue that it is correct.*

 

**Solution:** The Master theorem does not directly apply to this problem. We will try a different approach where we will first attempt to guess a solution. Firstly, note that we have $T(n) \geq 2T(n/2)$. Therefore, we have $T(n) \geq O(n)$. By a similar reasoning, we have $T(n) \leq 2T(4n/5)$. This gives $T(n) \leq O\left(n^{\log_{5/4} 2}\right)$. From these two inferences, we guess that $T(n)$ is probably of the form $an^b$ for some values of $a$ and $b$. To determine the value of $a$, we have $T(1) = 1 = a$. Therefore, we have $T(n) = n^b$. Now, to determine the value of $b$, we use the recurrence relation:

$$T(n) = n^b = T(3n/5) + T(4n/5) = \left(\frac{3}{5}\right)^b n^b + \left(\frac{4}{5}\right)^b n^b$$

The above equation is uniquely satisfied for $b > 0$ at $b = 2$ as the function $f(b) = (3/5)^b + (4/5)^b$ is decreasing in $b$. Therefore, the solution to the above recurrence is $T(n) = n^2$. Therefore, the answer to the $\Theta(n^2)$.