Note: Your TA may not get to all the problems. This is totally fine, the discussion worksheets are not designed to be finished in an hour. The discussion worksheet is also a resource you can use to practice, reinforce, and build upon concepts discussed in lecture, readings, and the homework.

1 Master Theorem

For solving recurrence relations asymptotically, it often helps to use the *Master Theorem*:

Master Theorem. If $T(n) = aT(n/b) + O(n^d)$ for $a > 0$, $b > 1$, and $d \geq 0$, then

$$T(n) = \begin{cases} O(n^d) & \text{if } d > \log_b a \\ O(n^d \log n) & \text{if } d = \log_b a \\ O(n^{\log_b a}) & \text{if } d < \log_b a \end{cases}$$

Note: You can replace O with Θ and you get an alternate (but still true) version of the Master Theorem that produces Θ bounds.

$d_{crit} = \log_b a$ is called the *critical exponent.* Notice that whichever of d_{crit} and d is greater determines the growth of $T(n)$, except in the case where they are perfectly balanced.

Solve the following recurrence relations and give a O bound for each of them.

(a) (i) $T(n) = 3T(n/4) + 4n$
(ii) $T(n) = 45T(n/3) + .1n^3$

(i) $a=3,\ b=4,\ \log_4 3 < 1$
$T(n) = O(n^{\cancel{\log_4 3}})$

(ii) $a=45,\ b=3,\ \log_3 45 > 3$
$T(n) = O(n^{\log_3 45})$

(b) $T(n) = 2T(\sqrt{n}) + 3$, and $T(2) = 3$.
Hint: Try repeatedly expanding the recurrence.

$T(n) = 2T(n^{\frac{1}{2}}) + 3$

$= 2\cdot(2\cdot T(n^{\frac{1}{4}}) + 3) + 3 = 2^2 \cdot T(n^{(\frac{1}{2})^2}) + (2^0 + 2^1)\cdot 3$

$= 2^k \cdot T(n^{(\frac{1}{2})^k}) + 3\cdot\sum_{i=0}^{k-1} 2^i$ (*)

In order to get the base case, $n^{(\frac{1}{2})^k} = 2$ so $(\frac{1}{2})^k \log n = 1 \Rightarrow k = \log\log n$

$T(n) = 2^{\log\log n} T(2) + 3\sum_{i=0}^{\log\log n - 1} 2^i$

$= 3\log n + 3\,O(\log n) = O(\log n)$

$3\,O(2^{\log\log n})$

(c) Consider the recurrence relation $T(n) = 2T(n/2) + n\log n$. We can't plug it directly into the Master Theorem, so solve it by adding the size of each layer.
Hint: split up the $\log(n/(2^i))$ *terms into* $\log n - \log(2^i)$*, and use the formula for arithmetic series.*

$\sum_{i=0}^{\log n} 2^i \left(\frac{n}{2^i}\cdot\log\frac{n}{2^i}\right) = \sum_{i=0}^{\log n} n\cdot\log\frac{n}{2^i}$

$= n\sum_{i=0}^{\log n} \log n - \log(2^i)$

$= n\left[\sum_{i=0}^{\log n}\log n - \sum_{i=0}^{\log n} i\cdot 1\right]$

$n\,O(\log n)$

$= n\left[\cancel{\log(\ldots)} - \frac{(1+\log n)\log n}{2}\right]$

$= \cancel{n\left[\frac{1}{2}\log n^2 + \log\right]}$

$= n\left[\log n \log^2 n - n\left(\frac{\log^2 n}{2} + \frac{\log n}{2}\right)\right]$

$= \frac{1}{2} n \log^2 n - \frac{1}{2}\cdot\log n$

$= O(n\log^2 n)$

2 Sorted Array

Given a sorted array A of n (possibly negative) distinct integers, you want to find out whether there is an index i for which $A[i] = i$. Devise a divide-and-conquer algorithm that runs in $O(\log n)$ time.

Consider about binary search.

①. ~~若~~ examine ~~A[~~ $A[\frac{n}{2}] = \frac{n}{2}$; return true.

② $A[\frac{n}{2}] > \frac{n}{2}$, ~~there~~ It is impossible that finding specific index in this range. examine the half,

③ $A[\frac{n}{2}] < \frac{n}{2}$, similian like step ②.

$O(\log n)$ time.

3 Quantiles

Let A be an array of length n. The boundaries for the k quantiles of A are $\{a^{(n/k)}, a^{(2n/k)}, \ldots, a^{((k-1)n/k)}\}$ where $a^{(\ell)}$ is the ℓ-th smallest element in A.

Devise an algorithm to compute the boundaries of the k quantiles in time $\mathcal{O}(n \log k)$. For convenience, you may assume that k is a power of 2.

Hint: Recall that QUICKSELECT(A, ℓ) gives $a^{(\ell)}$ in $\mathcal{O}(n)$ time.

The idea is finding the median of A, and then ~~spti~~ spilt it into two partition. Then recusively we will find the median of the two partition, spilt farther, and so on. Finally, when we find k-quantiles, we need do this logk time.

$$O(n+2\cdot n/2+2^2\cdot \frac{n}{2^2}+\cdots+2^{\log k}\cdot\frac{n}{2^{\log k}})=O(n\cdot\log k).$$

4 Complex numbers review

A *complex number* is a number that can be written in the rectangular form $a + bi$ (i is the imaginary unit, with $i^2 = -1$). The following famous equation (*Euler's formula*) relates the polar form of complex numbers to the rectangular form:

$$re^{i\theta} = r(\cos\theta + i\sin\theta)$$

In polar form, $r \geq 0$ represents the distance of the complex number from 0, and θ represents its angle. The n *roots of unity* are the n complex numbers satisfying $\omega^n = 1$. They are given by

$$\omega_k = e^{2\pi ik/n}, \qquad k = 0, 1, 2, \ldots, n-1$$

(a) Let $x = e^{2\pi i3/10}, y = e^{2\pi i5/10}$ which are two 10-th roots of unity. Compute the product $x \cdot y$. Is this a root of unity? Is it an 10-th root of unity?

What happens if $x = e^{2\pi i6/10}, y = e^{2\pi i7/10}$?

$x \cdot y = e^{\frac{2\pi i 8}{10}}$ so $x \cdot y$ ~~g~~ is still an 10-th roots of unity

(b) Show that for any n-th root of unity ω, $\sum_{k=0}^{n-1} \omega^k = 0$, when $n > 1$.

Hint: Use the formula for the sum of a geometric series $\sum_{k=0}^{n} \alpha^k = \frac{\alpha^{n+1}-1}{\alpha-1}$. It works for complex numbers too!

$$\sum_{k=0}^{n-1} \omega^k = \frac{\omega^n - 1}{\omega - 1} = \frac{0}{\omega - 1} = 0$$

(c) (i) Find all ω such that $\omega^2 = -1$.

$\because \omega^2 = -1$

$\therefore e^{\frac{2\pi i k}{2}} = -1$

(ii) Find all ω such that $\omega^4 = -1$.