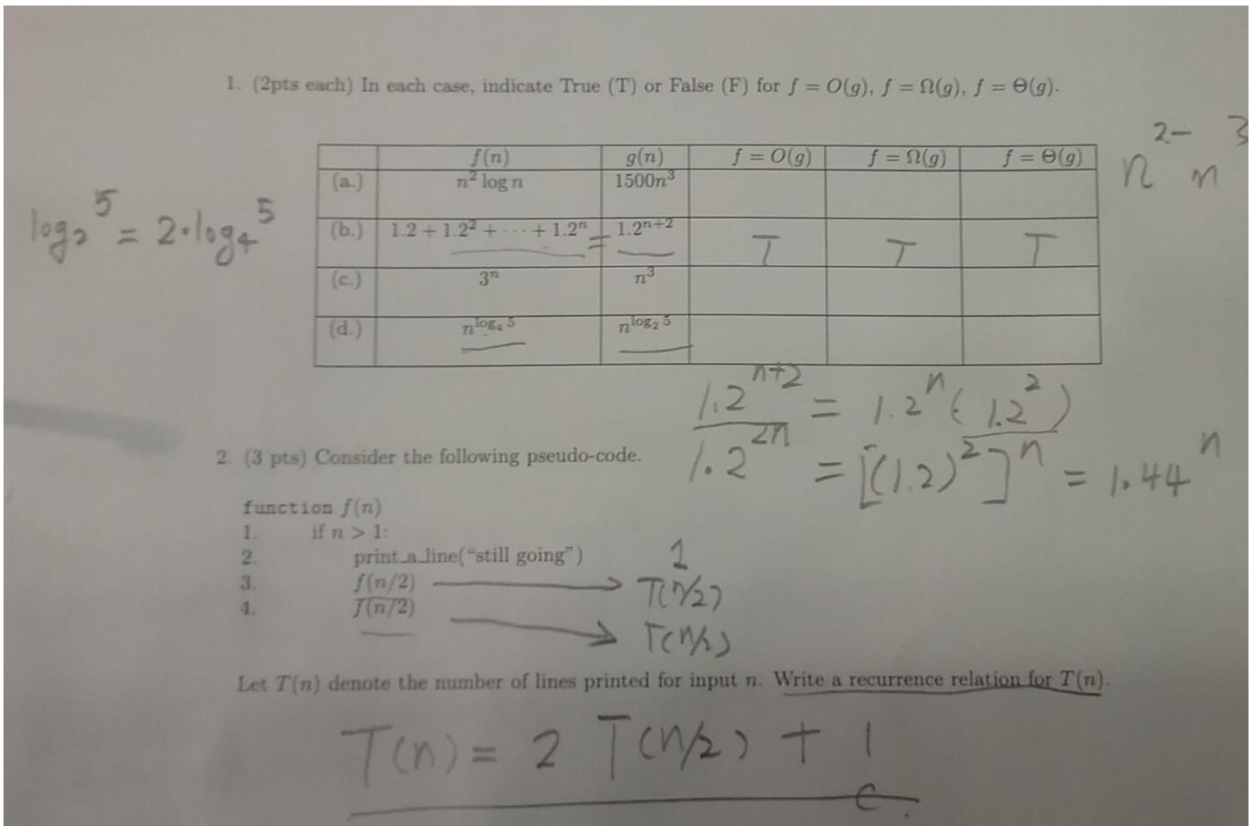
<u>Week 3 Tuesday Lecture Notes</u>

Prep: Review Slides for Closest Pair from Lecture 4.

Quiz Review:

Average: 10.5 out of 15 on Quiz.

Solutions for Quiz 1 are on Canvas.



1. (2pts each) In each case, indicate True (T) or False (F) for $f = O(g)$, $f = \Omega(g)$, $f = \Theta(g)$.

	$f(n)$	$g(n)$	$f = O(g)$	$f = \Omega(g)$	$f = \Theta(g)$
(a.)	$n^2 \log n$	$1500n^3$			
(b.)	$1.2 + 1.2^2 + \cdots + 1.2^n$	1.2^{n+2}	T	T	T
(c.)	3^n	n^3			
(d.)	$n^{\log_4 5}$	$n^{\log_2 5}$			

$\log_2 5 = 2 \cdot \log_4 5$

$n^2 \quad n^3$

$1.2^{n+2} = 1.2^n (1.2^2)$

$1.2^{2n} = [(1.2)^2]^n = 1.44^n$

2. (3 pts) Consider the following pseudo-code.

```
function f(n)
    if n > 1:
        print_a_line("still going")
        f(n/2)      ⟶ T(n/2)
        f(n/2)      ⟶ T(n/2)
```

Let $T(n)$ denote the number of lines printed for input n. Write a recurrence relation for $T(n)$.

$T(n) = 2T(n/2) + 1$

Start on Lecture 4: Page 7

-Finding the Closest Pair of a set of points in a 2D plane.

- Find the middle point L so half of the dots are on both sides. Then find the shortest length of points on both sides.

Observation: We only consider points that are closer than the two side pairs (the two pairs are 12 and 21. So the shortest so far is 12.

- O(nlogn). T(n) = 2T(n/2) + O(n).

Focusing on the Middle Strip:

- Any point that's outside the strip that is 12 lengths of both sides does not need to be considered. Because if it's outside the strip the length will be greater than 12.

Now we sort all the points inside the Strip.

- When we consider a point we only take account for points that are delta height and within the strip.
- Only need to consider a top band if you test points from bottom to top.

Cross Pair Comparison Code:

- List of points in the middle are already sorted in ascending order of y.
- Stepping through every node inside the strip, we find the minimum distance between the node.
- If a distance is shorter than Delta it will HAVE to be across the line L.
- You can ignore weather the points are on either side.
- Proof that if a point inside the strip has two points closer together than Delta than the algorithm will find it.

Algorithm: Closest-cross-pairs(M_y, δ)

$M_y = \{p_1, p_2, \dots, p_m\}$: the list of points in the middle strip sorted in increasing order of y

```
d_m = δ
for i = 1 to m − 1
    j = i + 1
    while p_j(y) − p_i(y) ≤ δ and j ≤ m
        d = D(p_i, p_j)
        d_m = min{d, d_m}
        j = j + 1
    end while
end for
Return d_m
```

at most 7 times

$O(m)$. $O(n)$

What if all the points are inside this Delta Strip and how will this affect runtime? (food for thought)

Claim: For any point: there can’t be 2 different points in the same cell. Therefore, you only need to compare with 7 points.

Page 11 of Lecture 4:

See Closest Pair Algorithm for Pseudo-Code for Implementation.

Closest Pair Algorithm

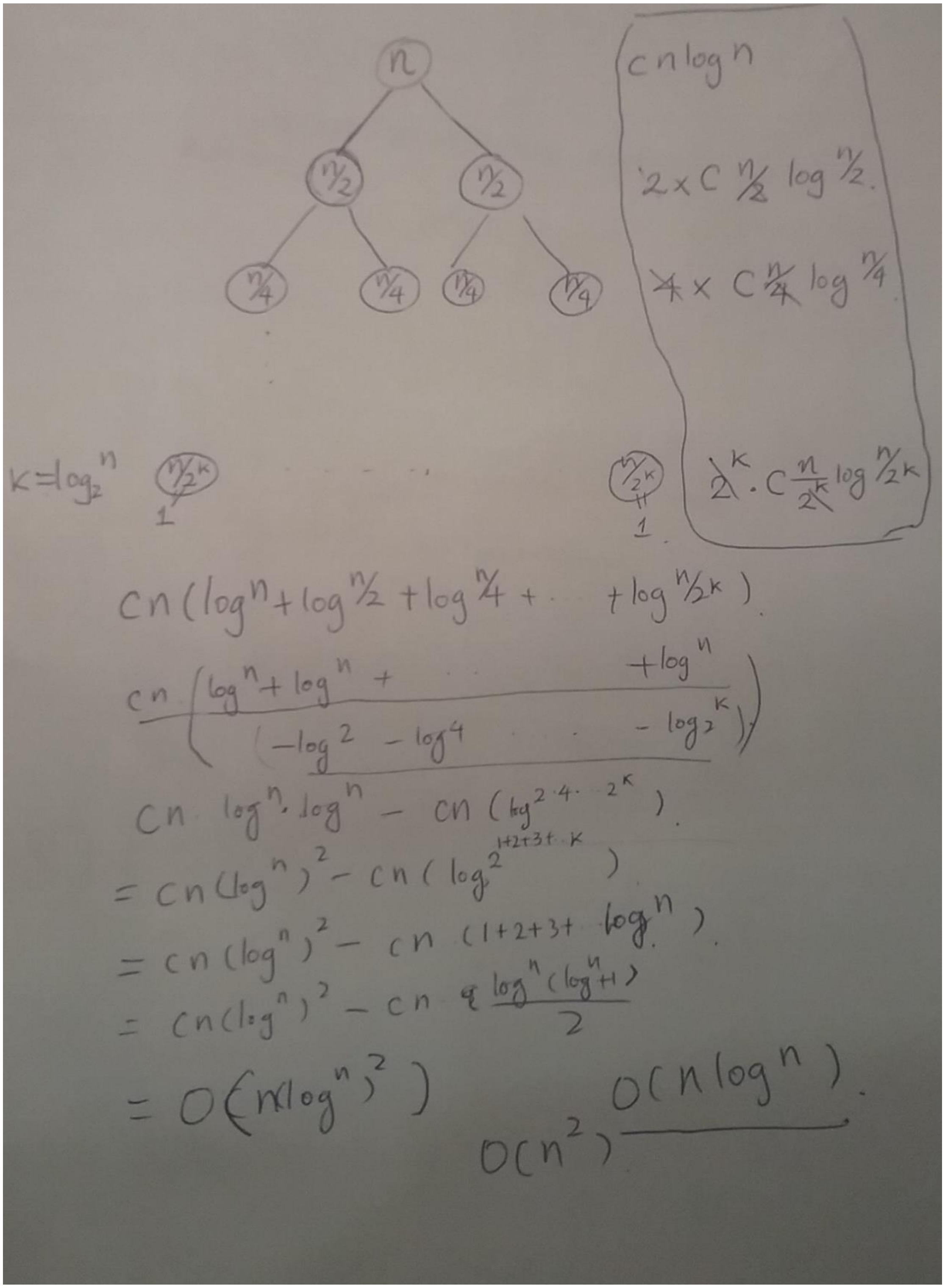
```
Closest-Pair(p_1, …, p_n) {
1. if n≤ 3
2.    compute and return the min distance
3. else
4.    Compute separation line L                    O(n log n)
5.    δ_1 = Closest-Pair(left half)                2T(n / 2)
6.    δ_2 = Closest-Pair(right half)
7.    δ   = min(δ_1, δ_2)

8.    Identify all points within δ from L          O(n)
9.    Sort them by y-coordinate into M_y           O(n log n)
10.   d_m = closest-cross-pair (M_y, δ)            O(n)

11.   return d_m.
}
```

Recurrence Relation Example for: 2T(n/2) + c*n*log (n)

Recurrence Picture



Currently we are at a big-O complexity of O (n log $(n)^2$)

Can we speed up the runtime and achieve O(n log n)?

Yes it is possible. To do this we don't sort points in strip from scratch, rather build them as we go. Saving the extra log n to end with a final efficient runtime of O(n log n)?

Ended PowerPoint Lecture 4.

Recap of Class:

- Reviewed Quiz 1 answers.
- Discussed Cross Pair Comparison.
- Recurrence Tree example.
- Explored how we can improve the runtime of the Cross Tree.

Next Time:

- **Review Question Set 2 for Wednesday's Recitation**
- **Recitation is between 5:45-6:45 Wednesday in the usual classroom.**
- **Quiz #2 will be at the end of class on Thursday.**
- **Assignment 1 about the Cross Pair problem with be further explained in class on Thursday.**

Week 3 Tuesday Lecture Notes

~Information composed by Notetaker Scott Russell for CS 325 **DAS** student