

# Lecture 24

CSE 331

Oct 29, 2021

# Please have a face mask on

## Masking requirement



*UR requires all students, employees and visitors – regardless of their vaccination status – to wear face coverings while inside campus buildings.*

<https://www.buffalo.edu/coronavirus/health-and-safety/health-safety-guidelines.html>

# Coding P1 due TODAY!

Fri, Oct 29	Counting Inversions  <sup>F19</sup>  <sup>F18</sup>  <sup>F17</sup> $x^3$	[KT, Sec 5.3] (Project (Problem 1 <b>Coding</b> ) in)
Mon, Nov 1	Multiplying large integers  <sup>F19</sup>  <sup>F18</sup>  <sup>F17</sup> $x^3$	[KT, Sec 5.5] (Project (Problem 1 <b>Reflection</b> ) in) <i>Reading Assignment: Unraveling the mystery behind the identity</i>
Wed, Nov 3	Closest Pair of Points  <sup>F19</sup>  <sup>F18</sup>  <sup>F17</sup> $x^3$	[KT, Sec 5.4]
Fri, Nov 5	Kickass Property Lemma  <sup>F19</sup>  <sup>F18</sup>  <sup>F17</sup> $x^3$	[KT, Sec 5.4] (Project (Problem 2 <b>Coding</b> ) in)
Mon, Nov 8	Weighted Interval Scheduling  <sup>F19</sup>  <sup>F17</sup> $x^2$	[KT, Sec 6.1] (Project (Problem 2 <b>Reflection</b> ) in)

# Group formation instructions

## Autolab group submission for CSE 331 Project

The lowdown on submitting your [project](#) (especially the [coding](#) and [reflection](#)) problems as a group on Autolab.

Follow instructions **EXACTLY** as they are stated

**The instructions below are for Coding Problem 1**

You will have to repeat the instructions below for EACH coding AND reflection problem on project on Autolab (with the appropriate changes to the actual problem).

## Form your group on Autolab

**Groups on Autolab will NOT be automatically created**

You will have to form a group on Autolab by yourself (as a group). Read on for instructions on how to go about this.

[Click to add notes](#)

# Preliminary grading rubric

note @401    my following **5** views

## Preliminary rubrics for reflections problems up

We have added preliminary grading rubrics for each reflection question:

<http://www-student.cba.buffalo.edu/~atr/bse331/fal21/projectreflection.html>

As noted in the page above, please keep in mind that in actual grading, we will use a grading rubric that expands on the preliminary grading rubric, i.e. you are NOT seeing the final rubric that will be used to grade your submissions.

We hope this preliminary grading rubric helps as y'all start working on the reflection questions.

[print](#)

## Preliminary Grading Guidelines

Below is a preliminary instantiation of the generic grading rubric above for (all ten parts of) Problem 1. In actual grading, we will use a grading rubric that expands on the preliminary grading rubric below.

- **Level 0**
  1. The authors did not respond with all 10 stakes; OR
  2. Answers may not be entirely relevant to the assignment.
- **Level 1**
  1. The authors did respond with all 10 stakes. Although, the responses may be underdeveloped; AND
  2. The authors clearly understand the questions, but have not demonstrated much effort in thinking through the different interests each stakeholder would have. Answers may seem perfunctory.
- **Level 2**
  1. The authors respond with all 10 stakes thoroughly and thoughtfully; AND
  2. The authors clearly demonstrate their grasp of the questions and the various perspectives each stakeholder might have on the same design; AND
  3. They demonstrate that what stakeholders' value differs depending on their own context.

# Questions/Comments?



# Rankings



# How close are two rankings?

Google compare rankings Search Advanced Search

Web Show options... Results 1 - 10 of about 23,700,000 for compare rankings. (0.30 seconds)

**Comparison Reviews**  
Angieslist.com Your neighbors' ratings on local service companies. Award-winning. Sponsored Link

**Ranking - Wikipedia, the free encyclopedia**  
In competition ranking, items that compare equal receive the same ranking number, and then a gap is left in the ranking numbers. The number of ranking...  
en.wikipedia.org/wiki/Ranking - Cached - Similar - Print

**A Comparison of Ranking Methods for Classification Algorithm Selection**  
is based on Spearman's rank correlation coefficient. To compare ranking methods, a combination of Friedman's test and Dunn's multiple com...  
www.springerlink.com/index.html?744744x70258.pdf - Similar - Print  
by PB Braudi • 2009 • Cited by 46 • Related articles

**College Football Ranking Comparison**  
College Football Ranking Comparison. Sunday, November 8, 2009 (127 Rankings) | Last Week | Archived | FBS | FCS | Summary | CSV Data | ...  
www.massseyratings.com/cf/compare.htm - Cached - Similar - Print

**College Basketball Ranking Comparison**  
Monday, April 6, 2009 (36 Rankings) --- Cor 923 810 923 898 819 888 888 777 639 899 880  
243 246 245 248 241 251 247 249 248 244 Ranking Violation % 254 ...  
www.massseyratings.com/cb/compare.htm - Cached - Print

**FIFA.com - Compare Teams**  
With FIFA.com you can compare the progress of up to four teams in the FIFA/Coca-Cola World Ranking. See how teams have risen and fallen since August 1993 ...  
www.fifa.com/worldfootball/ranking/compare/compareteams.html - Cached - Similar - Print

**GetEducated.com | Rate, Rank & Compare Online Colleges & Degrees**  
GetEducated.com's directory of online degrees provides rankings, ratings, and comparison tools to help you choose the best online degrees from the best ...  
www.geteducated.com/ - Cached - Similar - Print

**Compare your website rankings in Google Caffeine** < Advanced Web ...  
Aug 18, 2009 ... To help you compare your website rankings from Google with the rankings from Google Caffeine, we have created a quick step-by-step tutorial.

Web Images Videos Shopping News Maps More Mail Hotmail

bing compare rankings

ALL RESULTS 1-10 of 8,810,000 results - Advanced

**FIFA.com - Compare Teams**  
With FIFA.com you can compare the progress of up to four teams in the FIFA/Coca-Cola World Ranking. See how teams have risen and fallen since August 1993 and pinpoint their ...  
www.fifa.com/worldfootball/ranking/compare/compareteams.html - Cached page

**How to Compare Graduate School Rankings | eHow.com**  
Choosing a graduate school can be a stressful and confusing process for any student. By comparing graduate schools and their rankings a student can choose which one fits her ...  
By eHow Education Editor • Difficulty: Moderate • 0 posts  
www.ehow.com/how\_2051516\_compare-graduate-school-rankings.html - Cached page

**Ranking - Wikipedia, the free encyclopedia**  
Items that compare equal receive the same ranking number, which is the mean of what they would have under ordinal rankings. Equivalently, the ranking number of 1 plus the number of ...  
en.wikipedia.org/wiki/Ranking - Enhanced view

**Ranking Compare**  
Ask Google Yahoo Search | Web browser tool - Shows search results for a given keyword or phrase on the top three engines, Google, Yahoo and MSN.  
tools.essential-seo.com/ranking-compare - Cached page

**SchoolDigger.com - School Rankings, Reviews and More - Public and ...**  
Find the best elementary, middle, and high schools. Search for schools near any address, compare test scores, sort by school ranking, class sizes, and more using SchoolDigger.  
www.schooldigger.com - Cached page

**College and university rankings - Wikipedia, the free encyclopedia**  
The College and university rankings are a lists of universities and liberal arts colleges in higher education, an order determined by any combination of factors.  
International ... • Regional and national ... • Criticism (North America)  
en.wikipedia.org/wiki/College\_and\_university\_rankings - Enhanced view

**www.hospitalcompare.hhs.gov**  
www.hospitalcompare.hhs.gov/Hospital/Search/SearchCriteria.asp?version=default&browser=IE...

# Rest of today's agenda

Formal problem: Counting inversions

Divide and Conquer algorithm

# Problem definition on the board...



# Solve a harder problem

Input:  $a_1, \dots, a_n$

Output: LIST of all inversions

```
L =  $\phi$ 
for i in 1 to n-1
  for j in i+1 to n
    if  $a_i > a_j$ 
      add (i,j) to L
return L
```



Optimal for  
the listing  
problem

# Example 1: All inversions-- $(2i-1, 2i)$

2	1	3	4	6	5	7	8
---	---	---	---	---	---	---	---

Only check  $(i, i+1)$  pairs

Q1: Solve listing problem in  $O(n)$  time?

Q2: Recursive divide and conquer algorithm to count the number of inversions?

CountInv ( $a, n$ )

if  $n = 1$  return 0

if  $n = 2$  return  $a_1 > a_2$

$a_L = a_1, \dots, a_{\lfloor n/2 \rfloor}$

$a_R = a_{\lfloor n/2 \rfloor + 1}, \dots, a_n$

return CountInv( $a_L, \lfloor n/2 \rfloor$ ) + CountInv( $a_R, n - \lfloor n/2 \rfloor$ )

# Can be horribly wrong in general

CountInv (a,n)

if  $n = 1$  return 0

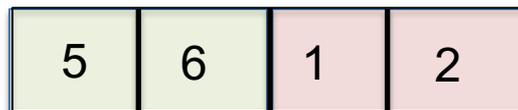
if  $n = 2$  return  $a_1 > a_2$

$a_L = a_1, \dots, a_{\lfloor n/2 \rfloor}$

$a_R = a_{\lfloor n/2 \rfloor + 1}, \dots, a_n$

return CountInv( $a_L, \lfloor n/2 \rfloor$ ) + CountInv( $a_R, n - \lfloor n/2 \rfloor$ )

Example where instance has non-zero (can be  $\Omega(n^2)$ ) inversions and algo returns 0?



All 4 "crossing" pairs are inversions

# Bad case: “crossing inversions”

CountInv (a,n)

if  $n = 1$  return 0

if  $n = 2$  return  $a_1 > a_2$

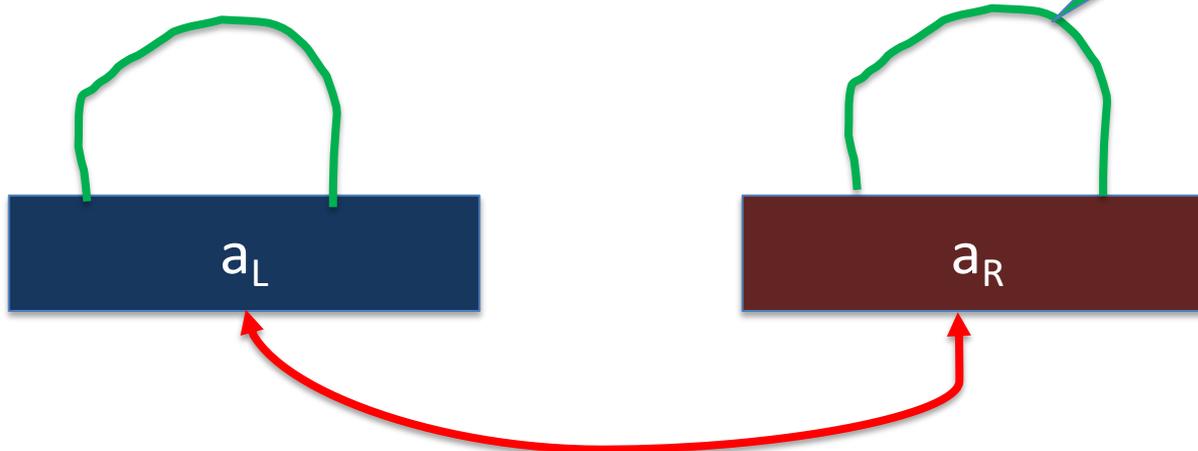
$a_L = a_1, \dots, a_{\lfloor n/2 \rfloor}$

$a_R = a_{\lfloor n/2 \rfloor + 1}, \dots, a_n$

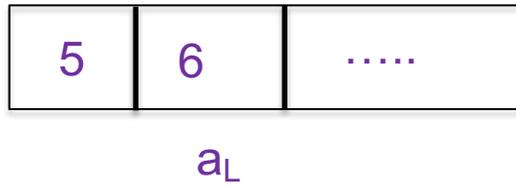
return CountInv( $a_L, \lfloor n/2 \rfloor$ ) + CountInv( $a_R, n - \lfloor n/2 \rfloor$ )

Yes!

Are  $a_L$   
and  $a_R$   
sorted?



# Example 2: Solving the bad case



$a_L$  is sorted

First element is  $a_L$  is larger than first/only element in  $a_R$

$O(1)$  algorithm to count number of inversions?

return size of  $a_L$

# Example 3: Solving the bad case

1

$a_L$

5 | 6 | .....

$a_R$

$a_R$  is sorted

First/only element is  $a_L$  is smaller than first element in  $a_R$

$O(1)$  algorithm to count number of inversions?

return 0

# Solving the bad case

First element of  $a_L$  is larger than first element of  $a_R$



$a_L$

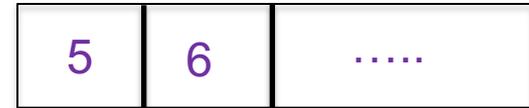


$a_R$

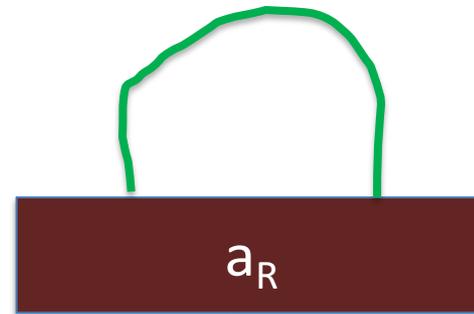
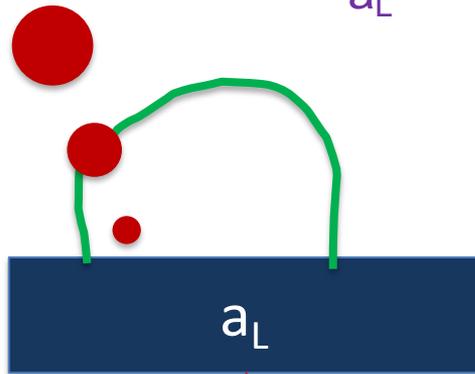
First element of  $a_L$  is smaller than first element of  $a_R$



$a_L$



$a_R$



Try to  
modify  
the  
MERGE  
algorithm

# Divide and Conquer

Divide up the problem into at least two sub-problems

Solve all sub-problems: Mergesort

Recursively solve the sub-problems

Solve stronger sub-problems: Inversions

“Patch up” the solutions to the sub-problems for the final solution

# MergeSortCount algorithm

Input:  $a_1, a_2, \dots, a_n$

Output: Numbers in sorted order+ #inversion

MergeSortCount(  $a, n$  )

If  $n = 1$  return ( 0 ,  $a_1$  )

If  $n = 2$  return (  $a_1 > a_2$ ,  $\min(a_1, a_2)$ ;  $\max(a_1, a_2)$  )

$a_L = a_1, \dots, a_{n/2}$       $a_R = a_{n/2+1}, \dots, a_n$

( $c_L, a_L$ ) = MergeSortCount( $a_L, n/2$ )

( $c_R, a_R$ ) = MergeSortCount( $a_R, n/2$ )

( $c, a$ ) = MERGE-COUNT( $a_L, a_R$ )

return ( $c+c_L+c_R, a$ )

$$T(2) = c$$

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

$O(n \log n)$  time

$O(n)$

Counts #crossing-inversions+  
MERGE

# MERGE-COUNT( $a_L, a_R$ )

$a_L = l_1, \dots, l_{n'}$

$a_R = r_1, \dots, r_m$

```
c = 0
```

```
i, j = 1
```

```
while i ≤ n' and j ≤ m
```

```
    if  $l_i \leq r_j$ 
```

```
        i ++
```

```
        add  $l_i$  to output
```

```
    else
```

```
        add  $r_j$  to output
```

```
        j ++
```

```
        c += n' - i + 1
```

```
Output any remaining items
```

```
return c
```



$a_L$



$a_R$



$a_L$



$a_R$