

# Syllabus Walkthrough

CSE 331

Aug 26, 2024

# Make sure you are on Piazza

**private note @4** 7 views

## Introduce Piazza to your students

### Post a Welcome Note!

In your first post on Piazza, welcome your students to their new class:

Students,

Welcome to Piazza! We'll be conducting all class-related discussion here this term. The quicker emails), the quicker you'll benefit from the collective knowledge of your classmates and instruct struggling to understand a concept—you can even do so anonymously.

-Atri Rudra

**Add Post**

### Include this blurb in your syllabus

This term we will be using Piazza for class discussion. The system is highly catered to getting you help fast and efficiently from classmates, the TA, and myself. Rather than emailing questions to the teaching staff, I encourage you to post your questions on Piazza. If you have any problems or feedback for the developers, email [team@piazza.com](mailto:team@piazza.com).

Find our class page at: <https://piazza.com/buffalo/fall2024/cse331/info>

Average Response Time: Special Mentions: Online Now | This Week:

<https://piazza.com/buffalo/fall2024/cse331/>

# Access to Autolab

CSE 331

Syllabus

Piazza

Schedule

Homeworks ▾

Autolab

Project ▾

Support Pages ▾

▶ channel

Sample Exams ▾

## Autolab

Details on Autolab, which will be used for all homework submissions in CSE 331.



### **A** Under Construction

This page is still under construction. In particular, nothing here is final while this sign still remains here.

### The main link

We will be using the UB CSE extension to [Autolab](#) for submission and (auto)grading of CSE 331 homeworks. You can access Autolab via <https://autolab.cse.buffalo.edu/>.

### Autolab location has changed!

If you had used Autolab in a semester before fall 2023 then you had used the previous version, which was at <https://autograder.cse.buffalo.edu/>. You can no longer access your old classes there so from Fall 2023 semester onwards, please use the updated location: <https://autolab.cse.buffalo.edu/>.

# Academic Dishonesty

All your submissions must be your own work

Penalty:

Minimum: A **grade reduction in course**

Possible: **F** (or higher penalty) if warranted

**YOUR** responsibility to know what is cheating, plagiarism etc.

If not sure, come talk to me

Excuses like “I have a job,” “This was OK earlier/in my country,” “This course is hard,” etc. **WON’ T WORK**

**I DO NOT HAVE ANY PATIENCE WITH ANY CHEATING :**  
**YOU WILL GET A GRADE REDUCTION IN THE COURSE**  
**FOR YOUR FIRST MISTAKE**

# Withdrawing a suspect submission

## Withdrawing a submission for academic integrity violation

Sometime mistakes can happen so you have the **option of withdrawing any of your Homework submission with 24 HOURS of the assignment deadline**. You can do this by sending Atri an email, e.g. by using the following template (thanks to [Oliver Kennedy](#) for providing us the template):

### Email template for withdrawing submission

Dear Dr. Rudra/Atri,

I wish to inform you that I have violated CSE 331 policies on my submission for Question X on Homeworks/Assignment N. I wish to withdraw my submission to preserve academic integrity.

J.Q. Student  
Person #12345678  
UBIT: jqstuden

Sincerely, J

On receiving the above email, I will assign J a 0 on Question X on Homeworks/Assignment N but disregard any Academic Integrity issues with the problematic submission. Note that J is not required to present any details on how they violated academic integrity.

- - - - -

# Use of ChatGPT is not allowed

AT

Why should I not use you to do my algorithms course homework?



While I can certainly help you understand concepts and provide explanations related to your algorithms course, there are several reasons why you shouldn't solely rely on me to do your homework:

1. **Academic Integrity:** Using external sources to complete your assignments without understanding the material yourself violates the principles of academic integrity. It's important to learn and demonstrate your knowledge honestly.
2. **Learning:** The primary goal of your course is to learn and understand the material. Relying solely on my answers without engaging with the material will hinder your learning and your ability to apply the concepts in real-world scenarios.
3. **Long-Term Retention:** Active engagement with the material, including struggling through problems and solving them on your own, enhances your understanding and retention of the content. Passive reading or copying solutions does not provide the same benefits.
4. **Exams and Practical Application:** If your course includes exams or projects that test your problem-solving skills, simply copying answers won't prepare you for those assessments. You need to be able to apply your knowledge independently.
5. **Conceptual Understanding:** Algorithms often involve complex concepts. Simply having an answer isn't enough; you need to understand why a particular approach works and how it relates to the broader context of the subject.

6. **Feedback and Improvement:** Making mistakes is an essential part of learning. When you attempt problems on your own, you can receive feedback from instructors or peers, which helps you identify areas for improvement.
7. **Career Preparedness:** Algorithms and problem-solving skills are often crucial in technical interviews and real-world applications. If you haven't practiced solving problems on your own, you might struggle in job interviews or when faced with practical challenges.
8. **Uniqueness of Assignments:** Homework assignments can vary in their requirements. If you rely on pre-generated solutions, you might miss out on tailoring your answers to the specific instructions of your assignment.
9. **Ethical Considerations:** While seeking help is acceptable, outsourcing the entire problem-solving process to an AI raises ethical concerns, especially if it's done to a degree that undermines the learning process.

It's okay to seek help and clarification on concepts, but make sure you're actively engaging with the material and attempting problems on your own. This will not only benefit your learning but also your overall academic and professional growth.

# In spirit of trust but verify

TAs will have access to a database of ChatGPT answers to cross check

It's no fun failing any of you but **use of ChatGPT will result in an F**

# Use of ChatGPT is not allowed

AT

Why should I not use you to do my algorithms course homework?



While I can certainly help you understand concepts and provide explanations related to your algorithms course, there are several reasons why you shouldn't solely rely on me to do your homework:

1. **Academic Integrity:** Using external sources to complete your assignments without understanding the material yourself violates the principles of academic integrity. It's important to learn and demonstrate your knowledge honestly.
2. **Learning:** The primary goal of your course is to learn and understand the material. Relying solely on my answers without engaging with the material will hinder your learning and your ability to apply the concepts in real-world scenarios.
3. **Long-Term Retention:** Active engagement with the material, including struggling through problems and solving them on your own, enhances your understanding and retention of the content. Passive reading or copying solutions does not provide the same benefits.
4. **Exams and Practical Application:** If your course includes exams or projects that test your problem-solving skills, simply copying answers won't prepare you for those assessments. You need to be able to apply your knowledge independently.
5. **Conceptual Understanding:** Algorithms often involve complex concepts. Simply having an answer isn't enough; you need to understand why a particular approach works and how it relates to the broader context of the subject.

6. **Feedback and Improvement:** Making mistakes is an essential part of learning. When you attempt problems on your own, you can receive feedback from instructors or peers, which helps you identify areas for improvement.
7. **Career Preparedness:** Algorithms and problem-solving skills are often crucial in technical interviews and real-world applications. If you haven't practiced solving problems on your own, you might struggle in job interviews or when faced with practical challenges.
8. **Uniqueness of Assignments:** Homework assignments can vary in their requirements. If you rely on pre-generated solutions, you might miss out on tailoring your answers to the specific instructions of your assignment.
9. **Ethical Considerations:** While seeking help is acceptable, outsourcing the entire problem-solving process to an AI raises ethical concerns, especially if it's done to a degree that undermines the learning process.

It's okay to seek help and clarification on concepts, but make sure you're actively engaging with the material and attempting problems on your own. This will not only benefit your learning but also your overall academic and professional growth.



# Coding jobs will be done by AI



## stacksort

In a [recent xkcd's](#) alt text, Randall Munroe suggested **stacksort**, a sort that searches StackOverflow for sorting functions and runs them until it returns the correct answer. So, I made it. If you like running arbitrary code in your browser, try it out.

Like (or hate) it? Comment on HackerNews

**stackoverflow\_sort**(  
Try a list of numbers, a string, a  
list of words or json.

);

Sort

**var output =**  
Output from the function.

;

output console

# Coding jobs will be done by AI

**MIT News**


ON CAMPUS AND AROUND THE WORLD

Browse

or

Search



 FULL SCREEN

Researchers have developed a flexible way of combining deep learning and symbolic reasoning to teach computers to write short computer programs. Here, Armando Solar-Lezama (left), a professor at CSAIL, speaks with graduate student Maxwell Nye.

Photo: Kim Martineau

## Toward artificial intelligence that learns to write code

Researchers combine deep learning and symbolic reasoning for a more flexible way

# It happened sooner than expected

**Support the Guardian**  
Fund independent journalism with \$5 per month  
[Support us](#) →

**The Guardian**

[News](#) [Opinion](#) [Sport](#) [Culture](#) [Lifestyle](#) [More](#) ▾

[The Guardian view](#) [Columnists](#) [Letters](#) [Opinion videos](#) [Cartoons](#)

**The Observer**  
ChatGPT

• This article is more than **4 months old**

## Programmers, beware: ChatGPT has ruined your magic trick

*John Naughton*



The generative AI tool can write code on request, making the specialist skill of programming open to everyone



Sat 1 Apr 2023 11.00 EDT

[f](#) [t](#) [e](#) **934**

Advertisement

 **United Healthcare**


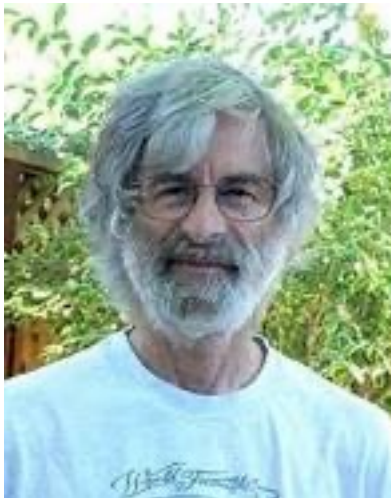
Complete a HouseCalls visit and earn up to a **\$15 reward**

[Learn more](#)

Must be enrolled in a qualifying UnitedHealthcare plan. HouseCalls may not be available in all areas. Rewards restrictions apply.

# So am I doomed?

There will still be room for high level *algorithmic* thinking!



European Association for  
Theoretical Computer Science


HOME ABOUT SEARCH CURRENT ARCHIVES

Home > No 125: June 2018 > **Lamport**

**If You're Not Writing a Program, Don't Use a Programming Language**  
*Leslie Lamport, Distributed Computing & Education Column by Juraj Hromkovic, Stefan Schmid*

*Today, programming is generally equated with coding. It's hard to convince students who want to write code that they should learn to think mathematically, above the code level, about what they're doing. Perhaps the following observation will give them pause. **It's quite likely that during their lifetime, machine learning will completely change the nature of programming. The programming languages they are now using will seem as quaint as Cobol, and the coding skills they are learning will be of little use. But mathematics will remain the queen of science, and the ability to think mathematically will always be useful.***

# Read the syllabus CAREFULLY!

CSE 331 Syllabus Piazza Schedule Homeworks ▾ Autolab Project ▾ Support Pages ▾  channel Sample Exams ▾

## CSE 331 Syllabus

### *Algorithms and Complexity*

Fall 2024

Time and location: **Mondays, Wednesdays** and **Fridays, 11:00-11:50am**, [KNOX](#)  104.



#### Under Construction

This page is still under construction. In particular, nothing here is final while this sign still remains here.

#### Please note

It is **your responsibility** to make sure you read and understand the contents of this syllabus. If you have any questions, please contact the instructor.

#### Acknowledgment

Once you have read the syllabus carefully, please fill in the Syllabus quiz on [Autolab](#). As an incentive for you to fill in this form, **you will not receive any feedback on your assignments till you successfully answer AT LEAST 18 out of the 20 questions in the quiz.** (You can attempt the quiz as many times as you want.) Note that in addition to this syllabus, the quiz will also ask questions based on the [homework policies](#).

# In spirit of trust but verify

## Syllabus Quiz

### Options

[View handin history](#)



Due: December 10th 2024, 11:59 pm EST (UTC -05:00)



Last day to hand in: December 10th 2024, 11:59 pm EST (UTC -05:00)



**No graded material will be handed back until you pass the syllabus quiz!**

### Academic Integrity

Question 1: Sharing my answers to this syllabus quiz with other 331 students

- Is OK if I do it to help out a friend
- It does not matter since there is no grade attached with it
- Is an academic integrity violation and should not be done
- Is an academic integrity violation but I can take the chance

# Accessibility Resources

Information included in the syllabus

In short, let me know and consult with Accessibility Resources

# Preferred Name

If you prefer using name diff from UB records


Let me know and we'll make a note of it.



# Lectures will be videotaped



# One Stop Shop for the Course

CSE 331 Syllabus Piazza Schedule Homeworks ▾ Autolab Project ▾ Support Pages ▾  channel Sample Exams ▾

## CSE 331

Fall 2024

<http://www-student.cse.buffalo.edu/~atri/cse331/fall24/index.html>

### Under Construction

This page is still under construction. In particular, nothing here is final while this sign still remains here.

### CSE 331 events

Today   Aug 18 – 24, 2024 ▾

 Print **Week** Month Agenda ▾

	Sun 8/18	Mon 8/19	Tue 8/20	Wed 8/21	Thu 8/22	Fri 8/23	Sat 8/24
5am							
6am							
7am							
8am							
9am							
10am							
11am							

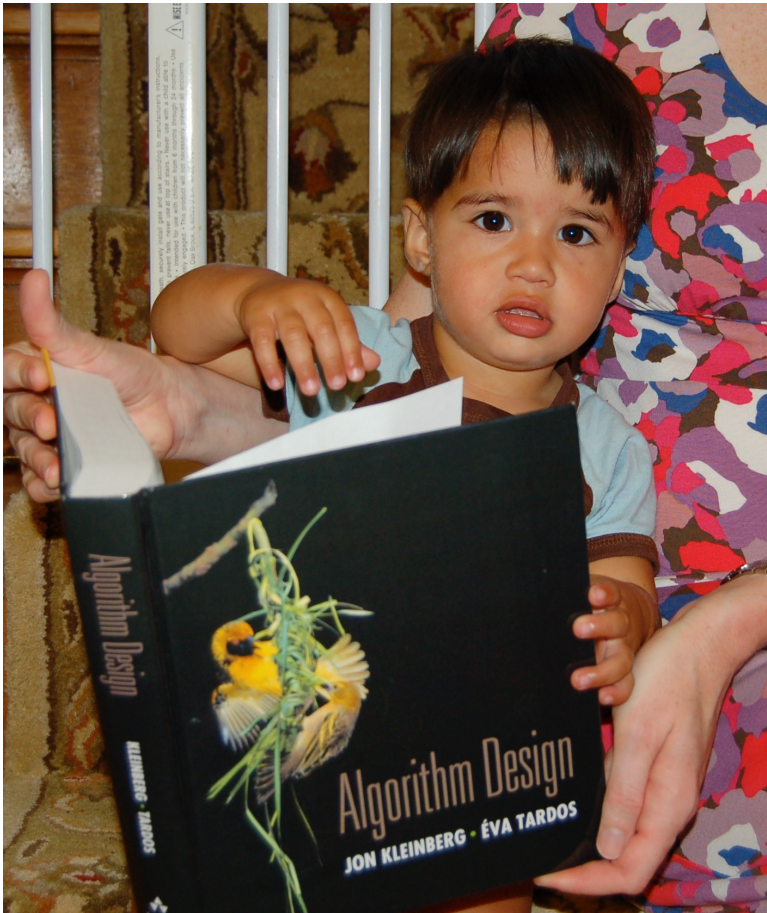
# Three things to remember

**WORK HARD!**

**DO NOT CHEAT!**

**READ CAREFULLY!**

# Wait.. What???



Make sure you follow submission instructions

Two most common ways  
of losing points

Make sure you read problem statements carefully

# Advice from 331 TAs

## CSE 331 Advice from TAs

Where students who took CSE 331 and became TAs share their experiences of how to fully utilize the class to your advantage. (And no, Atri did not pay them to say these things.)

<http://www-student.cse.buffalo.edu/~atri/cse331/support/advice/index.html>

### **A Under Construction**

This is a living document that will get updated over time. However, all the advice below is valid and you should pay attention to them!



## The class is structured to your advantage

### Utilize the before, during and after aspects of the course to their fullest.


Do the assigned readings before coming to class and if you get time even watch lecture videos from previous years. Atri will give you plenty of time during lecture to ask questions about the readings or the lecture itself. And of course get the most out of the assignments (Explained further below).

### The assignments are separated into different parts for your convenience.

#### Questions 1 and 2

For Q1 and Q2, think of the algorithm and proof ideas as things that go inside a header (.h) file. They are the high-level overview of how you are approaching the problem you don't

# More information on the quiz

CSE 331 Syllabus Piazza Schedule Homeworks ▾ Autolab Project ▾ Support Pages ▾  channel Sample Exams ▾

## CSE 331 Syllabus

### *Algorithms and Complexity*

Fall 2024

Time and location: **Mondays, Wednesdays** and **Fridays, 11:00-11:50am**, [KNOX](#)  104.

#### Under Construction

This page is still under construction. In particular, nothing here is final while this sign still remains here.

#### Please note

It is **your responsibility** to make sure you read and understand the contents of this syllabus. If you have any questions, please contact the instructor.

#### Acknowledgment

Once you have read the syllabus carefully, please fill in the Syllabus quiz on [Autolab](#). As an incentive for you to fill in this form, **you will not receive any feedback on your assignments till you successfully answer AT LEAST 18 out of the 20 questions in the quiz.** (You can attempt the quiz as many times as you want.) Note that in addition to this syllabus, the quiz will also ask questions based on the [homework policies](#).

# Autolab

AUTO LAB

You need to sign in before continuing.

<https://autolab.cse.buffalo.edu/>



Sign into Autolab

Students and Faculty:

[SIGN IN WITH SHIBBOLETH](#)

CSE IT Staff:

[MORE OPTIONS](#)



# You can submit the following now

» CSE 331: Algorithms and Complexity (f24)

## CSE 331: Algorithms and Complexity (f24)

Section: N

Number of grace days remaining: 0

 COURSE WEBSITE

 GRADEBOOK

## Assessments

Quiz

**Syllabus Quiz**

Start: Jul 29 at 8:11am | Due: Dec 10 at 11:59pm

If you were registered by 10am on Friday, Aug 23 you should be on Autolab



# Grading break-down

## Grading Policy

Here is the split of grades:

Course Component	% of grade
Project	10%
Homeworks	27%
Quizzes	3%
Exams	60%

## Changes from Fall 2023

A total score of  $< 20$  is an automatic F

# Questions/Comments?



# Pre-requisites

## Required (officially)

CSE 250, [CSE 191 or MTH 311] and MTH 142

At least a C- (this is recommended)

## Required (for practical purposes)

Comfort with proofs

Willingness to work hard!

# Critical Campus Resources

## Sexual Violence

UB is committed to providing a safe learning environment free of all forms of discrimination and sexual harassment, including sexual assault, domestic and dating violence and stalking. If you have experienced gender-based violence (intimate partner violence, attempted or completed sexual assault, harassment, coercion, stalking, etc.), UB has resources to help. This includes academic accommodations, health and counseling services, housing accommodations, helping with legal protective orders, and assistance with reporting the incident to police or other UB officials if you so choose. Please contact UB's Title IX Coordinator at 📞 716-645-2266 for more information. For confidential assistance, you may also contact a Crisis Services Campus Advocate at 📞 716-796-4399.

## Mental Health

As a student you may experience a range of issues that can cause barriers to learning or reduce your ability to participate in daily activities. These might include strained relationships, anxiety, high levels of stress, alcohol/drug problems, feeling down, health concerns, or unwanted sexual experiences. Counseling, Health Services, and Health Promotion are here to help with these or other issues you may experience. You can learn more about these programs and services by contacting:

### Counseling Services

120 Richmond Quad (North Campus), 📞 716-645-2720

### Health Services

4350 Maple Road (at Sweet Home Rd.) , 📞 716-829-3316

### Health Promotion

114 Student Union (North Campus), 📞 716-645-2837

# TA Office hours

## YOU decide!

The screenshot shows a Piazza Q&A page for CSE 331. The page is titled "TA office hours" and is a poll. The poll is currently closed, and the user is in the "poll admin panel". The poll description asks users to select all the time slots they will be able to attend for TA office hours. The poll is currently closed, and the user is in the "poll admin panel".

**poll admin panel** *visible only to instructors and creator of the poll*

Close the Poll Download Poll Stats Clone the Poll Hide Poll Results

**poll @8** stop following **12 views** Actions

### TA office hours

Please select all the time slots below that you will able to attend (even if for part of the time) for TA office hours. By default all office hours will be in person (but we might end up scheduling some as virtual depending on the TA availability). Recall that homeworks are due on Tuesdays at 11:30pm.

We will pick the top 29 choices at the end of Thursday August 29 (subject to TA availability) for the TA office hours.

Please note that the **TA office hours start Tuesday of 2nd week of class (i.e. from Tue, Sep 3).**

- Mondays, 9-10am
- Mondays, 10-11am
- Mondays, 12-1pm
- Mondays, 1-2pm
- Mondays, 2-3pm
- Mondays, 3-4pm
- Mondays, 4-5pm

Average Response Time: Special Mentions: Online Now | This Week:

# Recitations

Are on for this week!



Please stick to your recitation  
section

At least for the first month since all sections are full

# Exams

Mid term (two parts)

Mon, **Oct 7** and Wed, **Oct 9**. Usual place and time.

Final exam

Tue, **Dec 17**. Knox 109, **8:30-11:00am**

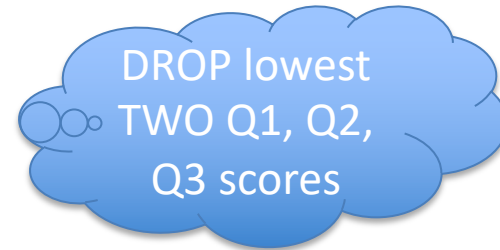


NOT the usual  
classroom



# The HW structure

Three questions



Q1 and Q2 are proof based while Q3 is programming

Q1 worth 50 points

The hard proof based Q2 and programming Q3 worth 25 points each

HWs due by 11:30pm on Tuesdays

# Allowed Sources

## Allowed sources

You can **ONLY** use the following sources for reference once you start working on the homework problems:

1. the Kleinberg-Tardos textbook,

### Other textbooks are not allowed

While you can use other textbooks (e.g. those listed in the [syllabus](#)) to better understand the lecture material, you **cannot** use them once you start working on the homeworks.

2. any material linked from this webpage or the CSE 331 piazza page (including any discussion in the Q&A section),

### One-click rule

When using webpages that are allowed as sources, you **cannot** click on link on that source. (Otherwise within a constant number of clicks one can reach any webpage one wants.)

3. specific *mathematical* result from a previous course,
4. anything discussed in the lectures, recitations and/or office hours and
5. any notes that you might have taken during class or recitation.

### Everything else is not allowed

Note that the above list covers all the allowed sources and **everything else is not allowed**. In particular, *YOU ARE NOT SUPPOSED TO SEARCH FOR SOLUTIONS ON THE*

# ... even for programming Q

CSE 331 Syllabus Piazza Schedule Homeworks ▾ Autolab Mini Project ▾ Support Pages ▾ Youtube channel

- All discussions and posts on [piazza](#).

## Basic programming references

### C++ Sources

- [cppreference.com](#) (and all pages within the website).

### Java Sources

- [Oracle Java Documentation](#) (and all pages within the website).

### Python Sources

- [Python 3.5.2 documentation](#) (and all pages within the website).



## Asymptotic Analysis

- [Big-O cheat sheet](#).

## Wikipedia Pages

Below are some approved Wikipedia pages (in addition to those that are already linked to in other pages in the [CSE 331 Fall 2018 web page](#)).

- [Gale Shapley algorithm](#).
- [DFS](#).
- [Dijkstra's Algorithm](#).
- [Prim's algorithm](#).

# Proof Idea vs. Proof Details

## Questions 1 and 2

For Q1 and Q2, think of the algorithm and proof ideas as things that go inside a header ( `.h` ) file. They are the high level overview of how you are approaching the problem; you don't have to be very technical here. For example, listing out all the steps in your algorithm, what proof technique are you using, what property of the algorithm are you induction on, etc.

Algorithm and proof details are the implementation inside the source ( `.cc` ) file. They are simply the detailed technical algorithm/ proof of the idea that was outlined.

### More on the idea vs details divide

Always start off with the ideas. Just smashing random keys on the keyboards won't get you anywhere with writing code and certainly would not help with proofs. In the real world, a user of your library doesn't care about the details; just wants to know how to use it. Similarly, in your proof and algorithm ideas, briefly explain what you're doing, how it works and why it should work. For example, if you're using contradiction in the proof details; just state that you use contradiction on a specific property (but do specify which property).

In the algorithm and proof details, be as detailed as you can be and try to avoid loopholes (more explained below).

# ~2 month long project!

## Individual Component

At the end of the project, you will rate your own and your other group member's contribution to the project. For more details, please see the [project page](#).

### Individual Component Grade

The individual component of your project will be worth 5% of your grade.

### Surveys to Individual component of project grade

Your survey scores will be converted into a fractional score  $\rho \in [0, 2]$ . We will reveal the exact algorithm after the surveys are submitted but roughly if everyone in the group did equal work (as reflected by the survey responses), then all group members will have  $\rho = 1$ . Otherwise, those that did more work will have a  $\rho$  value closer to 2 and those that did less will have  $\rho$  value closer to 0.

The survey part of the grade will be calculated as  $\rho \cdot \text{group score}$ , where **group score** is the sum of the coding and reflection components. If this score exceeds 5%, it will be capped at 6%.

### Acknowledgment

The development of this project was supported by a [Mozilla Responsible Computer Science award](#) . The support is gratefully acknowledged.

## Project has three parts

Your project will have three parts:

1. Do five **programming problems** that involves making tradeoffs between various choices among which some have ethical dimensions. This will be a group assignment.
2. Each programming question will be paired with (a series of) **reflection questions** that involves you writing down and reflecting on some of the design decisions you made in the corresponding programming problem. In particular, these questions will ask you to reflect on the societal and ethical implications of your decisions. This will also be a group assignment.
3. At the end of the project, each group member will fill in a **survey** rating their own and their other group member's contribution to the project.

# C++ vs Java/Python

## Use Java/Python if as you can

1. We recommend that you use a departmental server by `ssh` ing into it:

### Use a departmental server

Login to [one of the departmental servers accessible by students](#) and then run your code in there. Pick one of the servers that are described as `General compute server for short, interactive timeshare jobs`. `timberlake.cse.buffalo.edu` is one commonly used by students.

Unlike the VM options, you will need Internet to access the servers. Also unlike our first recommended option, the environment on departmental servers will not match the one on Autolab exactly but we do not expect this is to an issue.

If you still prefer to use your own system, we would recommend that you still test your code on a departmental server above before submitting to Autolab.

poll @7

stop following 14 views

Actions

## If you plan to use C++ for programming question

There will be one programming question per homework. You can submit your code in any of C++, Java and Python. If you plan to use Java or Python then you can ignore the rest of this post/poll.

If you plan to use C++, please first read the section titled *If you plan to use C++* in the [Autolab page](#) and then come back to this poll.

Please choose the option below on whether you would prefer to have dedicated office hours where you can get help setup things so that you can use the department linux servers to test your C++ code.

- I do not plan on using C++
- I plan to use C++ AND am comfortable running C++ from command line and NO office hours are needed
- I plan to use C++ AND would LIKE to attend an office hour for help with departmental server setup

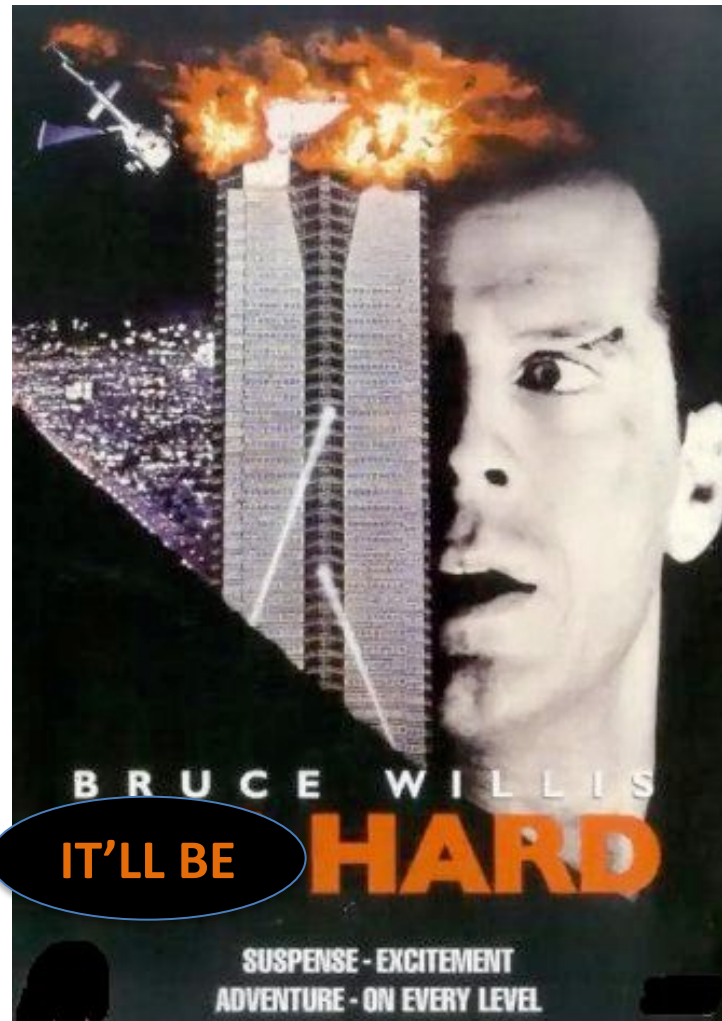
Please select one option

Submit

# Questions/Comments?

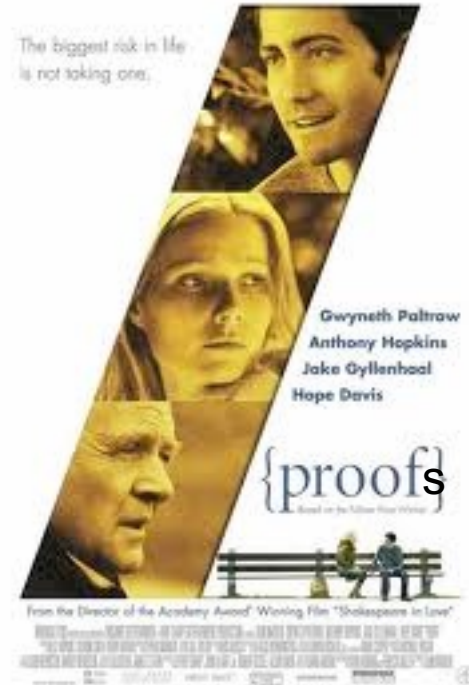


# Bit more about the course





# We'll do loads of



<http://www.impawards.com/2005/proof.html>

Writing down your thought process formally and precisely!

# An incorrect “proof”



# A more subtle incorrect “proof”

Brad Pitt had a beard



waleg.com

Every goat has a beard



animaldiversity.org

Hence, Brad Pitt is a goat.



# Why should we do proofs?

We will focus a lot on proofs in CSE 331. In this document I will motivate why doing proofs is good even though you might not do proofs for a living. While doing this, we will also go through examples of how to write algorithm ideas and details as well as proof ideas and details (which you will need to write in your homework solutions).

## Some reasons to do proofs

In this section, I will lay out some reasons why I think it is beneficial for you guys to do proofs. The first two are probably more along the lines of "if you do proofs for a living" situation. The rest of the reasons should be valid for all of you. I will try and make the reasons as concrete as possible: in the next section, we will consider algorithms for the specific problem of generating all permutations (recall that we [previously](#) had punted on designing an algorithm for this problem).

## Sometimes you might not have a choice

One of the easiest way to verify an algorithm idea you have is to code up the algorithm and then test it on some (say random) inputs. However, sometimes this might not be a choice. E.g. if you work on [Quantum Computing](#), then you do not have a quantum computer to run your quantum code on! So currently pretty much the only choice you have is to *prove* that your algorithm is indeed correct. For example, one of the crowning achievements of quantum computing is [Shor's algorithm](#) to computes the factors of large numbers efficiently on a quantum computer (that recall does not exist yet!). (You might also want to read [Scott Aaronson's high level description of Shor's algorithm](#).) The reason why [factoring large numbers](#) is important is that if one can solve this problem efficiently then one can break the [RSA cryptosystem](#). RSA is used everywhere (e.g. when you use your credit card online, RSA is used to make the transaction secure), so this is a big deal.

# CSE 331 Care package

## CSE 331 Care Package

Starting Fall 2019, CSE 331 will be assuming more background material was covered in CSE 250 (and CSE 191). In particular, starting Spring 2020, CSE 331 will assume a non-trivial coverage of proofs and other related material in CSE 191 and CSE 250. While we make this transition, this page collect materials that were covered in previous incarnations of CSE 331 but will no longer be covered going forward (this also includes material that are not covered in CSE 191/250). We hope that this page is helpful if you took the older version of CSE 191/250 or you took the equivalent courses in another school.

## The Topics

Below we collect the topics that we will not cover explicitly in CSE 331 (though some of these were covered as late as Fall 17 or Fall 18):

- [Reductions](#)
- [Asymptotic Notation](#)
- [Proof details of termination of Gale-Shapley algorithm](#)
- [Trees](#)
- [Topological Ordering](#)
- [Minimizing Maximum Lateness](#)



# A common complaint

Your examples in class look nothing like HW questions.

True because....



[zazzle.com](http://zazzle.com)

False because...

HWs and exams will test your **understanding** of the material



# To get an A in the class

Have to get at least 90.0000000000000000000000000000%

Rest graded on the curve **except anything below 20 is an F**

## Letter Grades

The letter grades will be determined based on a curve with the following two exceptions:

- To get an A in the course, you will *have* to obtain a total of 90.00% or more. This will not change (see the next callout for my reasoning).
- (New starting FA 24!) If you get a total < 20% then you will *automatically* get an *F*. Note that this does **not** mean that a total score of 20% automatically gets a D. The lower bound on total score for *D* will be set of at least 20% but the exact cutoff will be determined based on a curve.

## Few more thoughts on the grading scheme

My grading policy above is a bit non-standard in CSE courses so here are some follow up remarks:

- I will share the letter grade cutoffs from FA 23 on piazza by week 3 of classes to give y'all a sense of what the prior letter grade cutoffs were last time.
- I do get asked every year why I have an absolute scale for an A. Below is my reasoning (which you can feel free to disagree with but I wanted to share my thought process on this decision):
  - In my head the A grade signals complete mastery of the material and I wanted a scheme that would mean that a student getting an A in one semester would have gotten an any other semester as well. To have this hold true, I cannot curve an A. (If you are OK with the notion that an A needs to be absolute one could argue that 90 might not be the correct bar and that it should be lower (or higher). The reason to pick on my end was to pick a value that was reasonable to me and I still think my choice was fine.)
  - There have been suggestions to allocate an A to the top (or top  $x\%$ ) student(s) in class. This actually I'm vehemently opposed to because of the following reason:
    - Setting a "quota" of how many students will be an A (or any grade for that matter) actually puts more pressure on students. Specifically, my main reason to not do this is that this will incentivize students to not collaborate (in cases were collaboration is allowed). The way CSE 331 is setup, a student helping another student (again when it is allowed) does not hurt them in any way. However, having a quota will most likely discourage this collaborative effort.

# A cautionary tale...

When I was an undergrad

    Took algorithms as a sophomore

Understood all the lectures

Did not study outside of lectures

    (We had no homeworks)

Did decent on the mid-term

Nearly flunked the finals

Got a **C**



# Questions/Comments?



# How we will make 331



# What we'll strive to do

Help you with your questions and/or doubts

If need be, email us for time outside of regular office hours

We're not mind readers



If you need it, ask for help



# More chances to recover

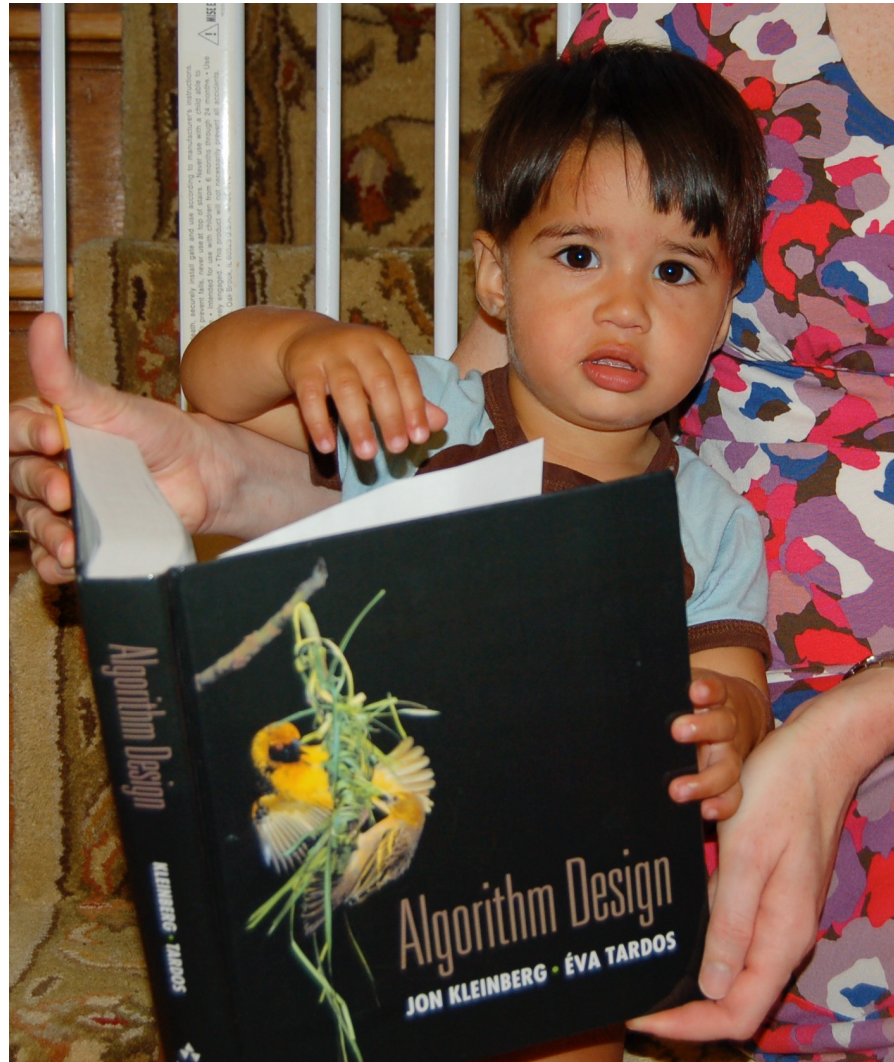
Lowest two Q1, Q2 and Q3 HW scores will be dropped

If you do better on the final exam than the mid-term exam

then only final exam score will count



# Follow the Textbook



# CSE 331 Support Page

This page contains certain webpages that students taking CSE 331 might find useful.



The material is roughly divided into two parts: one on (primarily mathematical) background material and one of common mistakes that students generally make.

## Disclaimer

Please note that this material is intended as a support material. It is not meant as a replacement for actually having taken background courses like CSE 116, 191 or 250 nor is this meant to be exhaustive. I'll try my best to make these as comprehensive as possible but that might take some time.

## Background material

CSE 331 will need a fair bit of math: most of which you must have seen earlier. However, if you have not used those material for a bit then you might be a bit rusty. The pages linked below are some notes that I wrote up that might help you refresh the material that you might have seen in CSE 116, 191 or 250. Also some of the

## Common Mistakes

Here we collect some common mistakes that students make in CSE 331 material (and sometimes more than once). The hope is to list these common pitfalls so that you can avoid them!

## Other Resources

Below we collect other 331 related material that do not neatly fall into the two left category:

- [Visualizing Algorithms](#).

<http://www-student.cse.buffalo.edu/~atri/cse331/support/index.html>

# The cautionary tale has a silver lining...



C in undergrad algorithms



Ph.D. in algorithms/complexity

The only way to do well is to work hard

