CSE 431/531: Algorithm Analysis and Design (Fall 2024)

Introduction I: Course Syllabus

Lecturer: Kelin Luo

Department of Computer Science and Engineering
University at Buffalo

Outline

1 Syllabus

Announcements: Academic Integrity Quiz

- Posted on Ublearns
- Should take $<$ 30 minutes, unlimited attempts
- Due Mon 09 Sep @ 11:59PM
- YOU MUST GET 100% TO PASS THE COURSE

CSE 431/531: Algorithm Analysis and Design

- Time & Location : Mon-Wed-Fri, 02:00pm - 02:50pm, Knox 110
- Instructor: Kelin Luo, kelinluo@buffalo.edu

- TAs: office hours, grading, Piazza questions, etc.
  - Xiaoyu Zhang (Head TA), zhang376@buffalo.edu
  - Ibrahim Bahadir Altun (Head TA), ialtun@buffalo.edu
  - Wen Zhang, wzhang59@buffalo.edu
  - Jason Niu, jasonniu@buffalo.edu
  - Mohammad Jakir Hossain, mh267@buffalo.edu
  - Sumaiya Islam Mouno, smouno@buffalo.edu
  - More TAs ...

- See Office Hours on Piazza.

CSE 431/531: Algorithm Analysis and Design

- Graders: grade and handle regrade requests on Piazza.
  - Kavitha Elizebeth Varughese
  - Xiaofeng Chen
  - Keshav Kumar Prabhakharan
  - Parag Shah

CSE 431/531: Algorithm Analysis and Design

- Course Webpage (contains Course Syllabus, news, schedule): https://cse.buffalo.edu/~kelinluo/teaching/cse431B:531B-fall24/index.html

- Ublearns: quiz, assignment submission, grading

- Please sign up course on Piazza via link https://piazza.com/buffalo/fall2024/cse431531b
  - announcements, asking/answering questions

Acknowledgement: The course design and information primarily draw inspiration from the Algorithm Analysis and Design course by Prof. Shi Li in Fall 2022 and Kelin Luo in Spring 2024.

CSE 431/531: Algorithm Analysis and Design

Introduces basic elements of the design and analysis of algorithms.

- Topics include asymptotic notations and analysis, algorithm frameworks, NP-completeness, and approximation algorithms.

CSE 431/531: Algorithm Analysis and Design

Introduces basic elements of the design and analysis of algorithms.

- Topics include asymptotic notations and analysis, algorithm frameworks, NP-completeness, and approximation algorithms.
- For each topic, beside in-depth coverage, we discuss one or more representative problems and algorithms.

CSE 431/531: Algorithm Analysis and Design

Introduces basic elements of the design and analysis of algorithms.

- Topics include asymptotic notations and analysis, algorithm frameworks, NP-completeness, and approximation algorithms.
- For each topic, beside in-depth coverage, we discuss one or more representative problems and algorithms.
- Learn discrete mathematics problem solving skills essential for computer scientists and engineers.

You should already have/know:

You should already have/know:

- Mathematical Background
  - basic reasoning skills, inductive proofs

You should already have/know:

- Mathematical Background
  - basic reasoning skills, inductive proofs
- Basic data Structures
  - linked lists, arrays
  - stacks, queues

You should already have/know:

- Mathematical Background
  - basic reasoning skills, inductive proofs
- Basic data Structures
  - linked lists, arrays
  - stacks, queues
- Some Programming Experience
  - e.g. Python, C, C++ or Java

You Will Learn

- Classic algorithms for classic problems
  - Huffman Code, Sorting, Shortest Paths, Minimum Spanning Tree, $\cdots$

You Will Learn

- Classic algorithms for classic problems
  - Huffman Code, Sorting, Shortest Paths, Minimum Spanning Tree, $\cdots$
- How to analyze algorithms
  - Correctness
  - Running time (efficiency)

You Will Learn

- Classic algorithms for classic problems
  - Huffman Code, Sorting, Shortest Paths, Minimum Spanning Tree, $\cdots$
- How to analyze algorithms
  - Correctness
  - Running time (efficiency)
- Meta techniques to design algorithms
  - Greedy algorithms
  - Divide and conquer
  - Dynamic programming
  - $\cdots$

You Will Learn

- Classic algorithms for classic problems
  - Huffman Code, Sorting, Shortest Paths, Minimum Spanning Tree, $\cdots$
- How to analyze algorithms
  - Correctness
  - Running time (efficiency)
- Meta techniques to design algorithms
  - Greedy algorithms
  - Divide and conquer
  - Dynamic programming
  - $\cdots$
- NP-completeness

Tentative Schedule

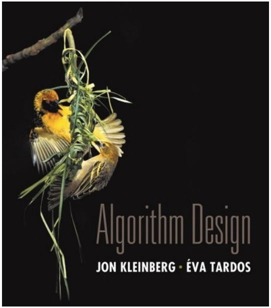
- 50 Minutes/Lecture × 42 Lectures

Introduction	3 lectures
Graph Basics	3 lectures
Greedy Algorithms	7 lectures
Divide and Conquer	6 lectures
Dynamic Programming	7 lectures
Graph Algorithms	8 lectures
NP-Completeness	5 lectures
Final Review	1 lectures
Mid-term Exam	2 lectures

Textbook

Textbook (Highly Recommended):

- Algorithm Design, 1st Edition, by *Jon Kleinberg* and *Eva Tardos*



Other Reference Books

- Introduction to Algorithms, Third Edition, *Thomas Cormen, Charles Leiserson, Rondald Rivest, Clifford Stein*

Reading Before and After Classes

- Highly recommended: read the correspondent sections from the textbook (or reference book or previous slides) before classes
  - Sections for each lecture can be found on the 2024 Spring course webpage.

Reading Before and After Classes

- Highly recommended: read the correspondent sections from the textbook (or reference book or previous slides) before classes
  - Sections for each lecture can be found on the 2024 Spring course webpage.
- Slides are posted on Piazza after the classes. After the lecture, you can review the updated slides on Piazza and access the course recordings in Brightspace.

Reading Before and After Classes

- Highly recommended: read the correspondent sections from the textbook (or reference book or previous slides) before classes
    - Sections for each lecture can be found on the 2024 Spring course webpage.

- Slides are posted on Piazza after the classes. After the lecture, you can review the updated slides on Piazza and access the course recordings in Brightspace.

- In last lecture of a major topic (Greedy Algorithms, Divide and Conquer, Dynamic Programming, Graph Algorithms), we will discuss in-class quiz problems.

Grading

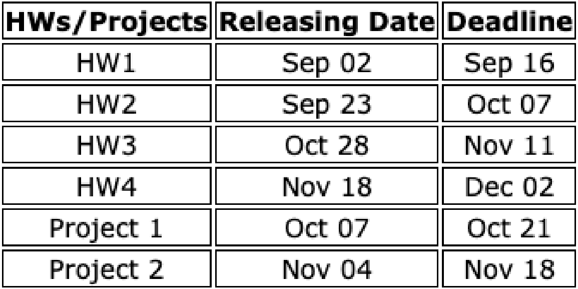
- 10% for participation
  - In-class quiz or Brightspace quizzes will be given randomly. (We choose the best 10 scores out of 12-15 quizzes.)
- 20% for theory homeworks
  - 5 points × 4 theory homeworks
    (typed PDF submissions, e.g. Microsoft Word, Latex.)
- 20% for programming projects
  - 10 points × 2 programming assignments
    (Programming: Python3 only)
- 20% for in-class mid-term exam (closed-book, closed-note)
- 30% for final exam (closed-book, closed-note)

Quiz

- You have $\geq$12 Quiz and the highest 10 scores will be counted.
- Ublearns Quiz (8-10 quizzes):
  - Each Quiz has at least 24 hours to complete it.
  - You have 2 attempts to complete each quiz and the highest score will be counted.
  - Ublearns auto-grade, no make-up quiz
- In-class Quiz (4-5 quizzes):
  - Random in-class quiz, and no late submissions will be accepted.
  - Grades will be released on UB Learns with an announcement on Piazza, no make-up quiz

Late Policy for (Written and Project) Assignments

- You have 2 weeks per assignment
  - Plan to start early and work throughout
  - 25% penalty per day late, up to 48 hours (Email submission to Instructor and Head TAs)
  - Grades will always be based on the LAST submission you make



HWs/Projects	Releasing Date	Deadline
HW1	Sep 02	Sep 16
HW2	Sep 23	Oct 07
HW3	Oct 28	Nov 11
HW4	Nov 18	Dec 02
Project 1	Oct 07	Oct 21
Project 2	Nov 04	Nov 18

Mid-term exam and Final exam

- In-Class Midterm I: Sep 27, Friday, 02:00-02:50PM
  - Content covered is roughly Weeks 1-5 in the syllabus (Introduction, Graph Basics, Greedy Algorithm)
- In-Class Midterm II: Nov 01, Friday, 02:00-02:50PM
  - Content covered is roughly Weeks 5-10 in the syllabus (Greedy Algorithm, Divide and Conquer, Dynamic Programming)
- Final exam: Dec 11, Wednesday, 07:15PM-10:15PM
  - Comprehensive, covering any topics from throughout the semester

Note: Check for conflicts. If you need accommodations, contact Accessibility Resources ASAP.

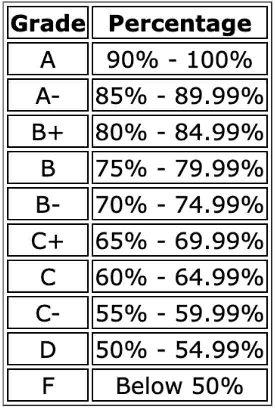
Re-grading

Question about the grading of any piece of work:

- First, consult with the grader and instructor on Piazza, following the regrade policy.
- Any questions about the grading of a piece of work must be raised within one week (or the specified regrade deadline on Piazza) of the date that the work was returned by the teaching assistant or the instructor.
- If you cannot resolve your questions on Piazza, then we recommend visiting office hours for further assistance.

Grading policy

- The following outlines the grade breakdown that will be utilized for assigning grades in the course.



Grade	Percentage
A	90% - 100%
A-	85% - 89.99%
B+	80% - 84.99%
B	75% - 79.99%
B-	70% - 74.99%
C+	65% - 69.99%
C	60% - 64.99%
C-	55% - 59.99%
D	50% - 54.99%
F	Below 50%

- Note that these ranges may be subject to adjustment at the end of the semester to address any inconsistencies or hardships that may arise.

Collaboration, AI, Extra Resources

- Do ...
  - Work together to brainstorm ideas
  - Explain concepts to each other
  - Include a list of your collaborators on all submitted work
- Do not ...
  - Write solutions when working together
  - Describe the details of solutions to problems or code
  - Leave your code in a place where it is accessible to another student

Collaboration, AI, Extra Resources

- Do ...
  - Work together to brainstorm ideas
  - Explain concepts to each other
  - Include a list of your collaborators on all submitted work
- Do not ...
  - Write solutions when working together
  - Describe the details of solutions to problems or code
  - Leave your code in a place where it is accessible to another student

When in doubt, ask a member of the course staff!

Resource Policy

- Do ...
  - Use materials provided by course staff (Piazza, Class, OH)
  - Use materials from the course lectures / hints
  - Cite all materials you reference for written work
  - Cite sources for all code you reference / copy

Resource Policy

- Do not ...
  - Reference random videos on YouTube that helped you solve the problem
  - Hire private tutors. If you have an actual tutor, contact instructor
  - Submit solutions you found on Google / Stack / Chegg / other source
  - Use of Artificial Intelligence Technologies like OpenAI's ChatGPT, Google Bard, and AI models within search interfaces like Google or Bing, etc.
  - Reference solutions found online
  - If you're not doing the work yourself, you're not learning

Resource Policy

- Do not ...
  - Reference random videos on YouTube that helped you solve the problem
  - Hire private tutors. If you have an actual tutor, contact instructor
  - Submit solutions you found on Google / Stack / Chegg / other source
  - Use of Artificial Intelligence Technologies like OpenAI's ChatGPT, Google Bard, and AI models within search interfaces like Google or Bing, etc.
  - Reference solutions found online
  - If you're not doing the work yourself, you're not learning

If you are caught using unauthorized resources, you get an F!

Other Ways to Get an F

- Work in a group by assigning each person to a problem
- Copying your friend's homework because you forgot
- Sharing your homework with your friend
  - "I have no way to know who did the work and who shared"
- Submitting work without citations
  - Citing outside work will help you avoid AI repercussions

Other Ways to Get an F

You are liable/punishable if someone else submits your work as their own.

Ways to Avoid an F

- Don't Cheat ... but we understand mistakes are made.

We will grant amnesty for any AI violation IF you tell us about it BEFORE we discover it

- Dear Dr. Luo,
  I wish to inform you that on assignment X, the work I submitted was not entirely my own. I would like to withdraw my submission from consideration to preserve academic integrity.
  J.Q. Student
  Person #12345678
  UBIT: jqstuden
- When we receive this email, student J would receive a 0 on assignment X, but would not receive an F for the course, and would not be reported to the office of academic integrity.

For Programming Projects

- Use Python version $\geq$3.4
- Need to implement the algorithms by yourself
- Can not copy codes from others or the Internet
- We detect similarity of programs, review the codes

For Programming Projects

- Use Python version $\geq$3.4
- Need to implement the algorithms by yourself
- Can not copy codes from others or the Internet
- We detect similarity of programs, review the codes

If cheating is found, you will get an "F" for the course. The case will be reported to the department.

Academic Integrity (AI) Policy for the Course

- minor violation:
  - 0 score for the involved homework/prog. assignment, and
  - 1-letter grade down
- 2 minor violations = 1 major violation
  - failure for the course: get an "F" for the course
  - further sanctions may include academic dishonesty mark on transcript or expulsion from university
- AI case will be reported to the department and university

- I. Notify the Student of the Concern
- II. Consult with the Student
- III. Decide the Sanction (Stop or AI violation)
- IV. Send the Decision Letter
- V. Assign the Grade (F with AD)

Course Sign up and Final exam date

- Last Day to Drop/Add a Course: Sep 03
- Resign Date: Nov 13

- Final exam: Dec 11, Wednesday, 07:15PM-10:15PM

Course Sign up and Final exam date

- Last Day to Drop/Add a Course: Sep 03
- Resign Date: Nov 13

- Final exam: Dec 11, Wednesday, 07:15PM-10:15PM
- Final exam conflict

Course Sign up and Final exam date

- Last Day to Drop/Add a Course: Sep 03
- Resign Date: Nov 13

- Final exam: Dec 11, Wednesday, 07:15PM-10:15PM
- Final exam conflict
  - Three or more final exams scheduled on the same day.

Course Sign up and Final exam date

- Last Day to Drop/Add a Course: Sep 03
- Resign Date: Nov 13

- Final exam: Dec 11, Wednesday, 07:15PM-10:15PM
- Final exam conflict
    - Three or more final exams scheduled on the same day.
    - Two final exams occurring at the same time.

Course Sign up and Final exam date

- Last Day to Drop/Add a Course: Sep 03
- Resign Date: Nov 13

- Final exam: Dec 11, Wednesday, 07:15PM-10:15PM
- Final exam conflict
  - Three or more final exams scheduled on the same day.
  - Two final exams occurring at the same time.
  - When a student's final exam occurs contemporaneously with his or her commencement ceremony for spring or summer conferral.

General Resources

Here are some of the University's available free resources:

- If you need help with writing, check UB Center for Excellence in Writing the Writing Support Services
- If you have issues with your device, the UB University Libraries provides access to computers, as well as equipment loans, see the Equipment Loans
- Your well-being is highly important, if you have any concerns, please check the Counseling Service

Accessibility Resources

- If you have any disability which requires reasonable accommodations to enable you to participate in this course, please contact the Office of Accessibility Resources in 60 Capen Hall, 716-645-2608 and also the instructor of this course during the first week of class.
- The office will provide you with information and review appropriate arrangements for reasonable accommodations, which can be found on the web at the Accessibility Resources.

Piazza Post Rule

You can post all questions related to lectures, quiz, and assignments on Piazza. Instructor posts announcement and course materials on piazza.

- For general questions about the course schedule, lectures, assignments, etc., please make your post visible to **everyone**.
- For personal inquiries regarding re-grades of homework or projects, please post and include both **me and the corresponding TA** in your post according to the Piazza regrade policy.
- For other personal queries related to this course, please post and include **me** in your post.

Email Rule

- Any email communications must come from your UB email account and include [CSE 431B/531B] in the subject line.
- All communications with course staff are expected to be professional.
- Late assignments will be accepted up to 2 days late (only by Email submission to instructor and two Head TAs).
- All other questions related to assignments and quizzes should be posted on Piazza.

Asking Questions

- First. . . check if the answer exists (syllabus, Piazza, course website)
- Then. . .
  - Ask in Piazza, or office hours
  - Come prepared, form the question carefully, many times you will answer your own question in the process!

Questions

Questions?

Questions

Questions?

Further questions, please post on Piazza!