Data Structures & Algorithms
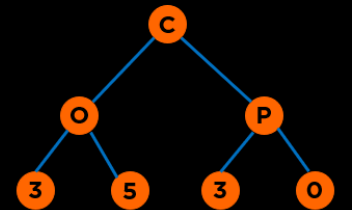
COP 3530 : Fall 2025

Data Structures & Algorithm

About Me: Amanpreet Kapoor

- **Educator**

- **CS Education Researcher**

- **Mentor**

- **Software Engineer**

- **Lifelong Learner**

kapooramanpreet@ufl.edu

Course Staff

Tavienne
Teaching Lead

Jackie
Communications Lead

Shae

Sampada

Clarissa

Justin

Adrian

Cole

James

Declan

Ananya

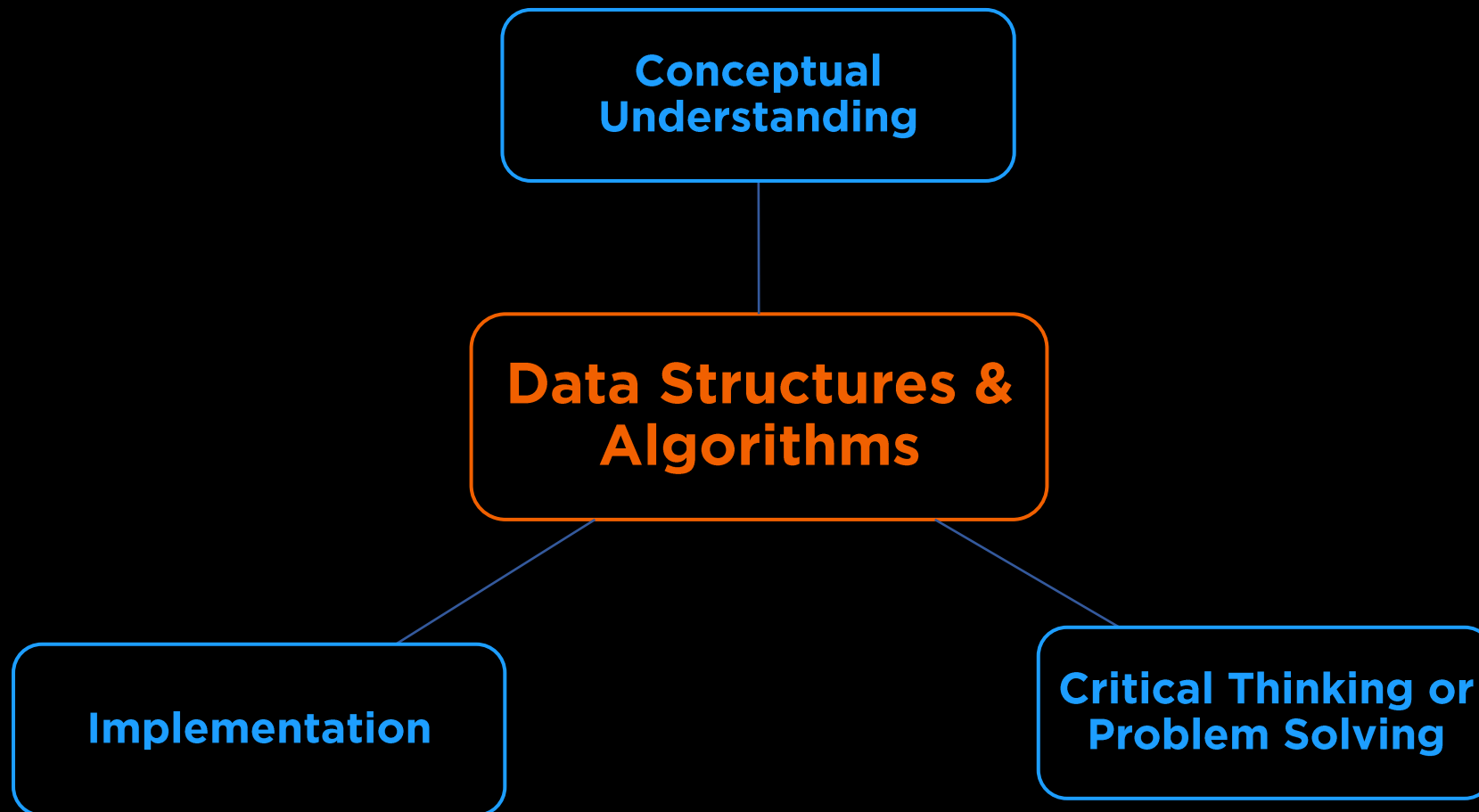# Learners: Let's Get to Know You

Go To Menti.com

Code: **7423 2722**

Course Objectives

What is this Course About?

This course covers algorithm development using

- **pseudo languages**

- **basic program structures**

- **program design techniques**

- **storage and manipulation of basic data structures**

- **3 Credit Hours**

What is this Course About?

Categories of Data Structures

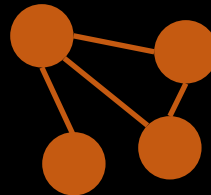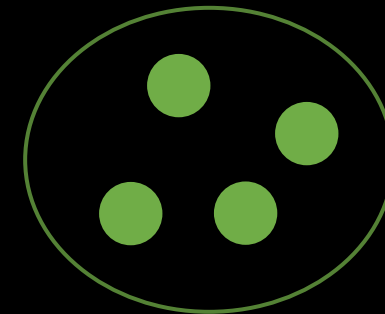| Linear Ordered	Non-linear Ordered	Not Ordered
Lists	Trees	Sets
Stacks	Graphs	Tables/Maps
Queues		

Categories of Algorithms

Brute Force	Divide & Conquer	Greedy	Dynamic Programming
Selection Sort	Binary Search	Minimum Spanning Tree	Knapsack
Bubble Sort	Merge Sort	Shortest Paths	Fibonacci
Insertion Sort	Quick Sort		
NP Complete Problems			
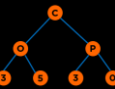
Logistics & Policies

Format

Action Items by Monday morning

Watch lectures synchronously on Tue and Thu in person or live via Zoom.
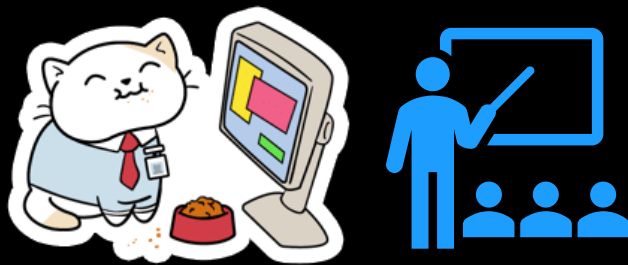
Attend discussions on Wed.
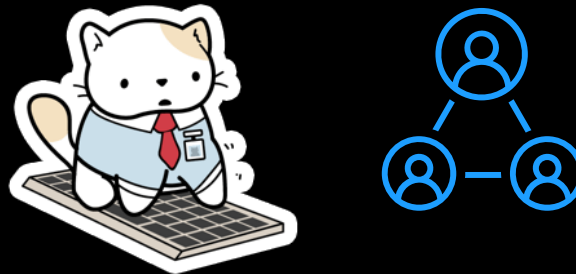
1. Conceptual Quiz due on Sat.
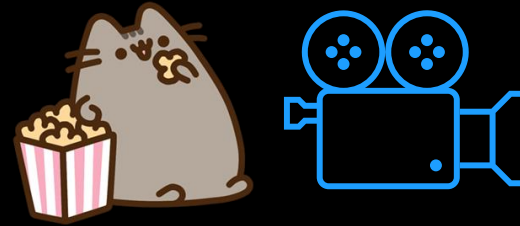2. Programming Quiz due on Sat.

Communication

Slack

- Everything!
  - for all question related to the course
  - for communicating with peers
  - one-to-one communication with me

- Use Appropriate Tags

- Response in < 48 business hours

Office Hours

- Tuesday, 4-5 pm

- Friday, 1-2 pm

- By Appointment (24 hours in advance)

Communication

Email

Fine for

- Personal
- Accommodations
- Emergencies

Email

Not okay for

- Questions regarding logistics
- Questions on coding
- Fixing bugs

Communication

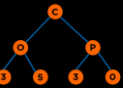~~Canvas Messages~~ ~~Phone Call~~

Debugging

- Students should visit the course staff during scheduled office hours for help and provide context for help.

- Debugging requests for projects/quiz questions must first go through the TAs or peer mentors. This is strongly encouraged given we have a large class and several of you might have similar questions.

- If your problem is not fixed, then start a conversation with both the Instructor and the TA/Peer mentor who you asked for help. Debugging requests to the Instructor as a Slack direct message or an email will be ignored if you do not follow the above protocol.

Grading

Modality	Assignment	% of Final Grade
Individual	Programming / HTG Quizzes (drop two lowest scores)	10%
	Conceptual quizzes (drop two lowest scores)	10%
	Exam 1	19%
	Exam 2	19%
	Project 1	12%
	Project 3	12%
	Class participation (drop four lowest scores)	3%
	Discussion participation (drop four lowest scores)	3%
Collaborative	Project 2 (Individual or Group: 2a & 2b)	12%
Individual	Extra Credit Opportunities	Up to 3%
		Total: 103%

Total assessments: 26 excluding extra credit and class/discussion participation

Timeline

Week	Dates		Topic	Deadlines
1	21-Aug	24-Aug	Course Introduction/Algorithm Analysis	
2	25-Aug	31-Aug	Algorithm Analysis / List, Stacks, & Queues	Q_1
3	1-Sep	7-Sep	Trees	Q_2
4	8-Sep	14-Sep	Balanced Trees 1	Q_3
5	15-Sep	21-Sep	Balanced Trees 2	Q_4
6	22-Sep	28-Sep	Heaps & Priority Queues	P_1
7	29-Sep	5-Oct	Exam 1 (October 2)	Q_5, E_1
8	6-Oct	12-Oct	Sorting	P_{2a}, Q_6
9	13-Oct	19-Oct	Sets, Maps, & Hashing	
10	20-Oct	26-Oct	Graphs 1	Q_7
11	27-Oct	2-Nov	Graphs 2	Q_8
12	3-Nov	9-Nov	Greedy Algorithms	P_{2b}
13	10-Nov	16-Nov	Exam 2 (November 13)	Q_9, E_2
14	17-Nov	23-Nov	Dynamic Programming	Q_{10}
15	24-Nov	30-Nov	Thanksgiving Break	
16	1-Dec	3-Dec	Complexity Theory	P_3
Legend: Q_N = Quiz N, P_N = Project N, E_N = Exam N				

Programming Language

Default (Project 1 & 3, Edugator, Quizzes):

C++14

Compilation command:

g++ -std=c++14 -Werror -Wuninitialized -o EXECUTABLE_NAME YOUR_FILE.cpp

Project 2:

Any Language

Tools

Compiler

- G++

- Edugator/Gradescope
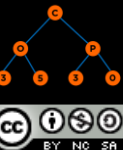
IDE

- <u>OnlineGDB</u>

- Visual Studio Code

- Clion

Tools

Directory Structures

- src – includes .h and .cpp files

   (You can use lib/ too for separating interfaces - .h)

- test – contains test files

- build – contains executables (e.g. .exe)

- Know basic commands on compilation through command line:

  - cd, ls, mv, g++, pwd …

Textbook (Optional)

- Data Structures and Algorithm Analysis in C++
  - Mark Allen Weiss
  - Fourth edition, 2014, ISBN 9780132847377

- OpenDSA Book
  - https://opendsa-server.cs.vt.edu/OpenDSA/Books/Everything/html/index.html

Feedback

- COP 3530 Feedback Form

- Bug Bounty Program

  - Quiz is not accessible due to a locked module
  - Typo in one of the quizzes/project descriptions
  - Incorrect solution in a certain test case
  - An accessibility bug such as no headings in documents for screen-readers
  - the algorithm has an off-by-one error

Expectations

- We want you to focus on
  - Correctness
  - Clean, readable, tested, and documented code
  - Secondary focus on optimization

- The course will not cover
  - Mathematical Proofs
  - Design Patterns
  - Competitive Programming

Expectations

Conceptual Quizzes and Exams:

- Work independently

- No discussion at the conceptual level

- You are allowed to
    - discuss solutions after the due date and late days have passed.

Expectations

Programming Quiz questions:

- Work independently or as a pair (Select a buddy in Week 1).

- The buddy must remain the same for the entire course.

- The course staff can help you on syntax issues, but we will not help you with semantic and logic issues for quizzes.

- Cite that you worked with a buddy in Line 1 of your code and add their name.

- You are allowed to

    - discuss solutions after the due date and late days have passed.

    - search for C++ syntax or refer to definitions of standard functions in the C++ library. For example, using the documentation listed here is fine: https://www.cplusplus.com/reference/ or https://en.cppreference.com/w/

Academic Dishonesty

Project 1 and 3:

- Work independently
- Discussion at the conceptual level is fine if you are stuck with no sharing/viewing of code

- You are allowed to
  - discuss conceptually without discussing any code with a peer provided you cite the peer with who you discussed it. Such discussions should be held on a whiteboard using explanation figures/pseudo-codes or through talking.
  - discuss solutions after the due date and late days have passed.
  - search for C++ syntax or refer to definitions of standard functions in the C++ library. For example, using the documentation listed here is fine: https://www.cplusplus.com/reference/ or https://en.cppreference.com/w/

Academic Dishonesty

Project 2 and Edugator ungraded questions:

- It is fine to collaborate with peers. You must make sure you are not blindly copy-pasting another student's code. Also, you must cite the peer you worked with at the code level or conceptually.

Academic Dishonesty and GenAI policy

- Sharing, copying, "borrowing" code, viewing another student's work, or plagiarism of any form is academic dishonesty. You may not view or share external code, search for solutions online, use genAI tools such as ChatGPT to generate entire functions/programs, use AI tools to write/translate reports, or have someone else complete your work.

- AI tools may be used for learning concepts or subtasks, but not to produce solutions for quizzes, exams, or projects. If GenAI tools are used for learning, you must cite the tool, model, and prompt.

- Penalty

  - reported to the Honor Court
  - E grade for the course

- Regret Clause:
  - If you submit an assignment, in which you engaged in some of the unacceptable practices, you may bring it to my attention by emailing me and withdrawing your assignment within 48 hours of the submission. If you do so, I will assign a failing grade of 0 for the assignment, and you will not be reported to the administration.

Request for Extensions

- Any request for assessment extensions should be backed by official documentation (e.g., from a medical professional, etc.) and should be sent over email. Requests without documentation will be ignored. The following reasons deem ineligibility for extensions or regrades:

  - Failure to submit on Canvas by due date/late date. Example: If a quiz is due at 11:59 pm, and you send the file at 12:01 am, the file will not be graded. Note that most assessments are open for 4 or more days and you must start early and submit it on time. In case you miss an assignment, treat it as a learning activity and avoid that in the future.
  - Forgetting to turn in an assessment on time.
  - File naming issues or feedback issues on projects. Gradescope gives you feedback and please read it and fix your file and resubmit. Note that you have unlimited attempts on Gradescope for everything and we will not grade your files if you do not adhere to instructions on file submissions and/or if the Gradescope scores your file to 0. It is your responsibility to read the feedback and fix your code. In case you miss reading the feedback, treat it as a learning activity and avoid that in the future.

Acknowledgements

Cheryl Resch
Lecturer,
Dept. of Engineering Education,
University of Florida
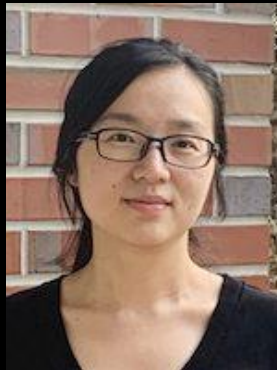
Lisha Zhou
Lecturer,
Dept. of Engineering Education,
University of Florida

References

- Books/Notes
    - Dr. Sartaj Sahni
    - Dr. James Aspen
    - Dr. Mark Weiss
    - Dr. Clifford Shaffer
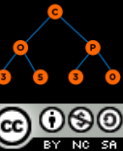   - OpenDSA
    - Dr. Cathy Hughes
- Videos Authority
    - Dr. Josh Hug
    - MIT OCW 6.006: Dr. Erik Demaine and Dr. Srini Devadas
    - Dr. Robert Sedgewick

- Videos Youtube
    - HackerRank: Data Structures
    - HackerRank: Algorithms
    - Back To Back SWE
    - MyCodeSchool
    - Abdul Bari
- GeeksforGeeks

Walkthrough

Walkthrough

- Canvas

- OpenDSA

- Discord

- Edugator

- Gradescope

Questions