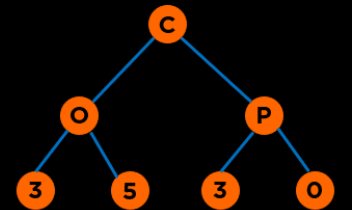


# Data Structures & Algorithms

COP 3530 : Spring 2023



# About Me: Amanpreet Kapoor

- Educator
- CS Education Researcher
- Mentor
- Software Engineer
- Lifelong Learner



[kapooramanpreet@ufl.edu](mailto:kapooramanpreet@ufl.edu)

# Course Staff



Robin  
(Teaching Lead)



Andrew  
(Infrastructure Lead)



Dustin



Julia



Amay



Olivia



Rutvi



Prayuj



Nitin



Emily



Eugene



Eric



Benjamin



Sara

Kevin Allen  
Yair Temkin  
Antonio Villarreal

Haohui Bao  
Andrew Kennedy  
David Dexter

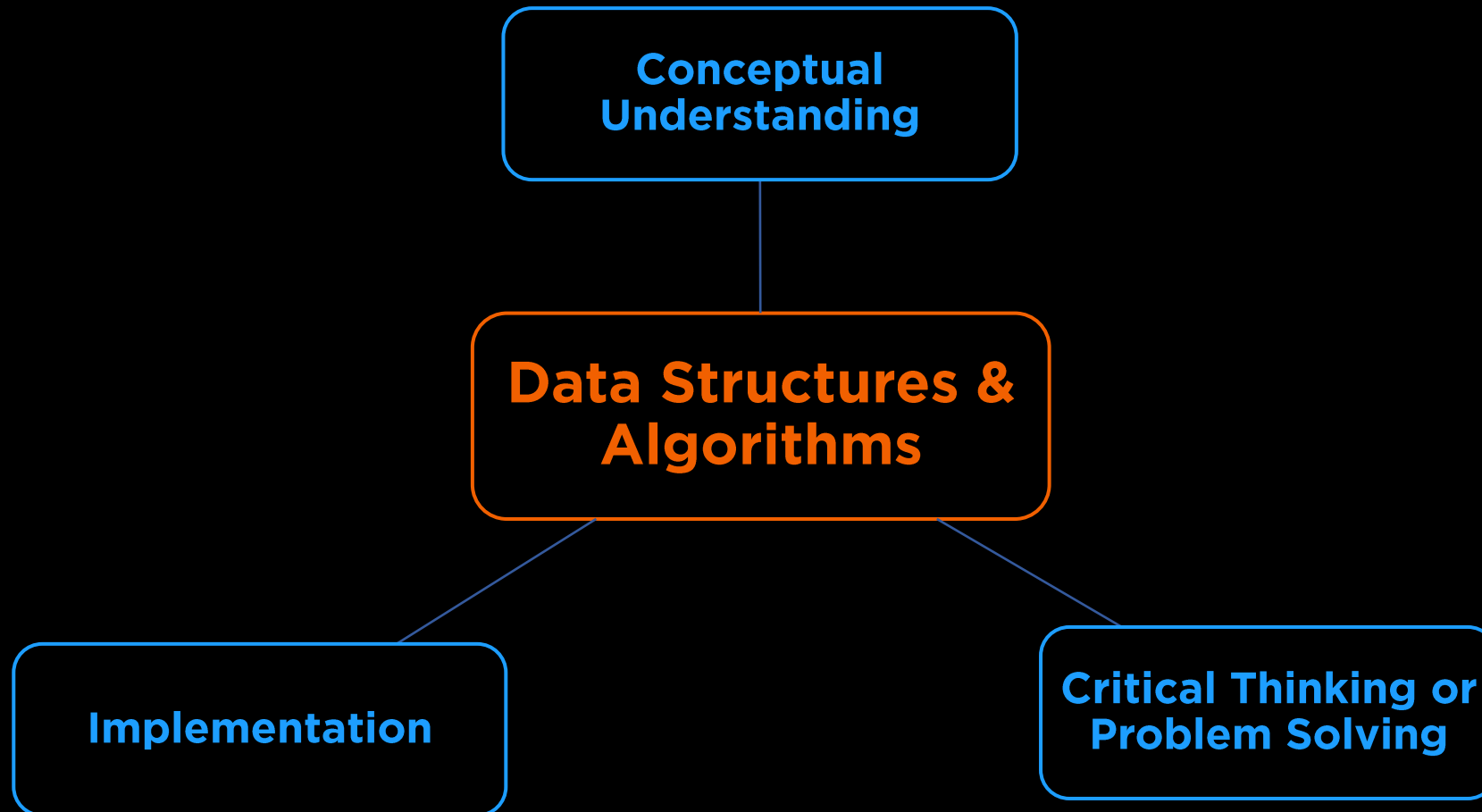
# Course Objectives

# 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

Lists

Stacks

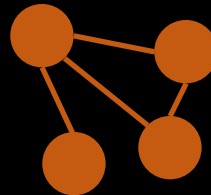
Queues



Non-linear Ordered

Trees

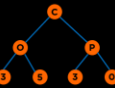
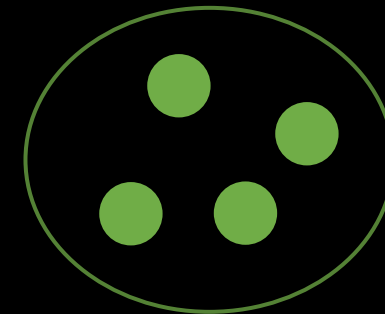
Graphs



Not Ordered

Sets

Tables/Maps



# Categories of Algorithms

**Brute Force**

**Selection Sort**

**Bubble Sort**

**Insertion Sort**

**NP Complete Problems**

**Divide & Conquer**

**Binary Search**

**Merge Sort**

**Quick Sort**

**Greedy**

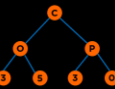
**Minimum Spanning Tree**

**Shortest Paths**

**Dynamic Programming**

**Knapsack**

**Fibonacci**





# Logistics & Policies

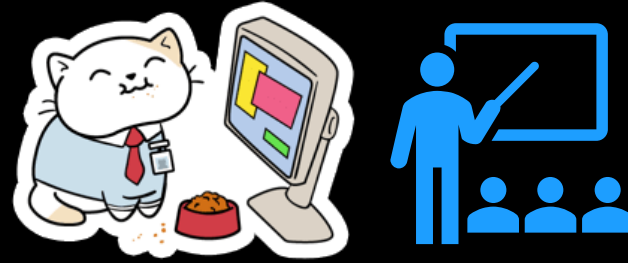
# Changes this semester

1. There will be no quizzes due during the week of Exam 1 and 2 or Project 1 and 2.
2. Total quizzes are reduced from 11 to 10.
3. Final exam grade weight is reduced by 2%.
4. Project grades are increased by 2% for each of the three projects.
5. Students can pair program on the weekly programming quizzes with one student that they select in week 1.

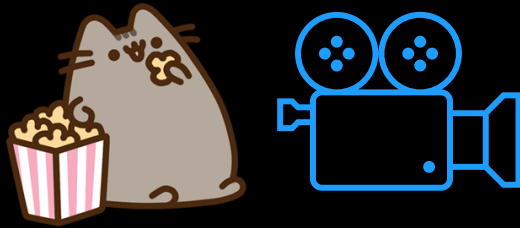
# Format



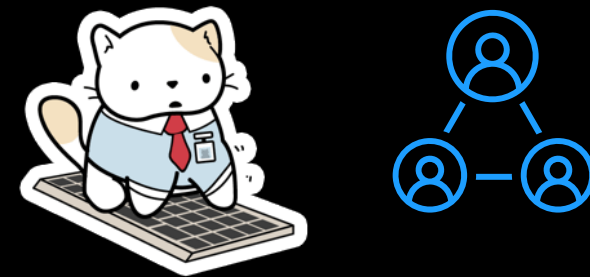
Action Items by  
Monday morning



Attend discussions on  
Tue.



Watch lectures asynchronously or  
attend synchronously on MWF.



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



- Wed 4:30 pm – 5:30 pm
- Fri 4:30 pm – 5:30 pm
- By Appointment (24 hours in advance)

# Communication



## Email

### Fine for

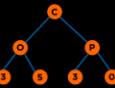
- **Personal**
- **Accommodations**
- **Emergencies**



## 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

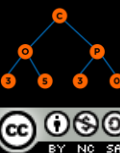
Modality	Assignment	% of Final Grade
Individual	Programming / HTG Quizzes (drop two lowest)	12%
	Conceptual quizzes (drop two lowest)	10%
	Exam 1	20%
	Exam 2 (Cumulative)	20%
	Project 1	14%
	Project 2	12%
Collaborative	Final Project (Individual or Group: 3a & 3b)	12%
Individual	Extra Credit Opportunities & Bug Bounty Program	Up to 2%
		Total: 102%

Total assessments: 26 excluding extra credit and class participation



# Timeline

Week	Dates		Topic	Deadlines
1	9-Jan	15-Jan	Course Introduction / Algorithm Analysis	
2	16-Jan	22-Jan	Algorithm Analysis / List, Stacks, & Queues	$Q_1, Q_2$
3	23-Jan	29-Jan	List, Stacks, & Queues / Trees & Traversals	$Q_3$
4	30-Jan	5-Feb	Trees & Traversals / Balanced Trees 1	$Q_4$
5	6-Feb	12-Feb	Balanced Trees 1 and Project 1 Overview	
6	13-Feb	19-Feb	Balanced Trees 2	$Q_5$
7	20-Feb	26-Feb	Heaps & Priority Queues / Sorting	$P_1$
8	27-Feb	5-Mar	Exam 1	$E_1$
9	6-Mar	12-Mar	Sorting / Sets, Maps, & Hashing	$P_{3a}, Q_6$
10	13-Mar	19-Mar	Spring Break	
11	20-Mar	26-Mar	Sets, Maps, & Hashing / Graphs 1	$Q_7$
12	27-Mar	2-Apr	Graphs 1 / Graphs 2	$Q_8$
13	3-Apr	9-Apr	Graphs 2 / Greedy Algorithms	$P_2$
14	10-Apr	16-Apr	Exam 2	$E_2$
15	17-Apr	23-Apr	Greedy Algorithms / Dynamic Programming	$Q_9$
16	24-Apr	26-Apr	Dynamic Programming / Complexity Theory	$P_{3b}, Q_{10}$
Legend: $Q_N$ = Quiz N, $P_N$ = Project N, $E_N$ = Exam N				



# Programming Language

**Default (Project 1 & 2, Stepik/EduGator, Quizzes):**

**C++14**

Compilation command:

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

**Project 3 or Final Project:**

**Any Language**

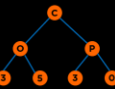
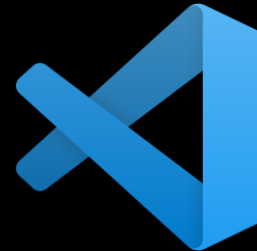
# Tools

## Compiler

- G++
- Stepik/EduGator/Gradescope

## IDE

- OnlineGDB
- Visual Studio Code
- Clion

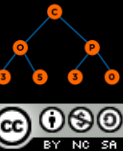


# Tools

## 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` ...

<https://cplusplus.com/forum/beginner/261697>  
<https://stackoverflow.com/questions/2360734/whats-a-good-directory-structure-for-larger-c-projects-using-makefile>



# Textbook (Optional)

- **Data Structures and Algorithm Analysis in C++**

- Mark Allen Weiss
- Fourth edition, 2014, ISBN 9780132847377

- **OpenDSA Book**

- <https://opensa-server.cs.vt.edu/OpenDSA/Books/Everything/html/index.html>

# Feedback

- COP 3530 Feedback Form
- Bug Bounty Program : Upto 1% EC in Range 0.2-1% per Bug
  - 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

Quiz questions on Canvas 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.
  - 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/>



# 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 2:

- 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 3 and Stepik 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

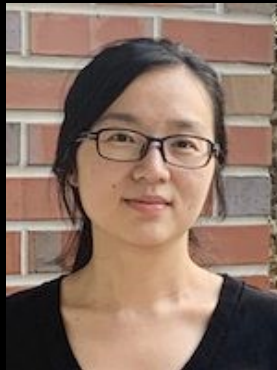
- Sharing/copying, “borrowing” of code structure, looking at code from another student or providing such code, and plagiarism, in addition to other dishonest behaviors, are all considered to be academic dishonesty.
- No information regarding the project 1 and 2, quiz, and exam solutions may be shared by students. We strongly encourage that if you have doubts, visit the course staff in-office hours. Looking at any piece of your peer’s code, sharing files, searching for solutions found online, or using someone else to code your solution is strictly prohibited.
- Penalty
  - zero on that assignment and a two-letter final grade decrement for a first offense
  - E grade for second offence
  - For both offenses, you will be reported to the Honor Court

# 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
- OpenDSA
- [Dr. Cathy Hughes](#)

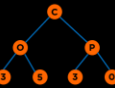
- Videos Authority

- [Dr. Josh Hug](#)
- [MIT OCW 6.006](#): Dr. Erik Demaine and Dr. Srinivas Devadas
- [Dr. Robert Sedgewick](#)

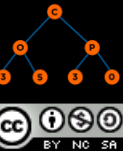
- Videos Youtube

- [HackerRank: Data Structures](#)
- [HackerRank: Algorithms](#)
- [Back To Back SWE](#)
- [MyCodeSchool](#)
- [Abdul Bari](#)

- GeeksforGeeks



# Walkthrough



# Walkthrough

- Canvas
- OpenDSA
- Slack
- Stepik/Eduigator
- Gradescope



# Questions

