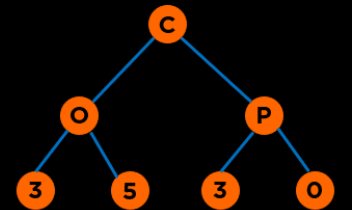


Data Structures & Algorithms

COP 3530 : Summer 2023



About Me: Amanpreet Kapoor

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



kapooramanpreet@ufl.edu

Course Staff



Robin
(Teaching Lead)



Andrew Penton
(Infrastructure Lead)



Dustin



Julia



Amay



Olivia



Rutvi



Prayuj



Nitin



Emily



Eugene



Eric



Benjamin



Sara



Andrew
Kennedy



Antonio



Kevin



Haohui

Yair Temkin
David Dexter

Learners: Let's Get to Know You

Go To Menti.com

Code: 4103 7678

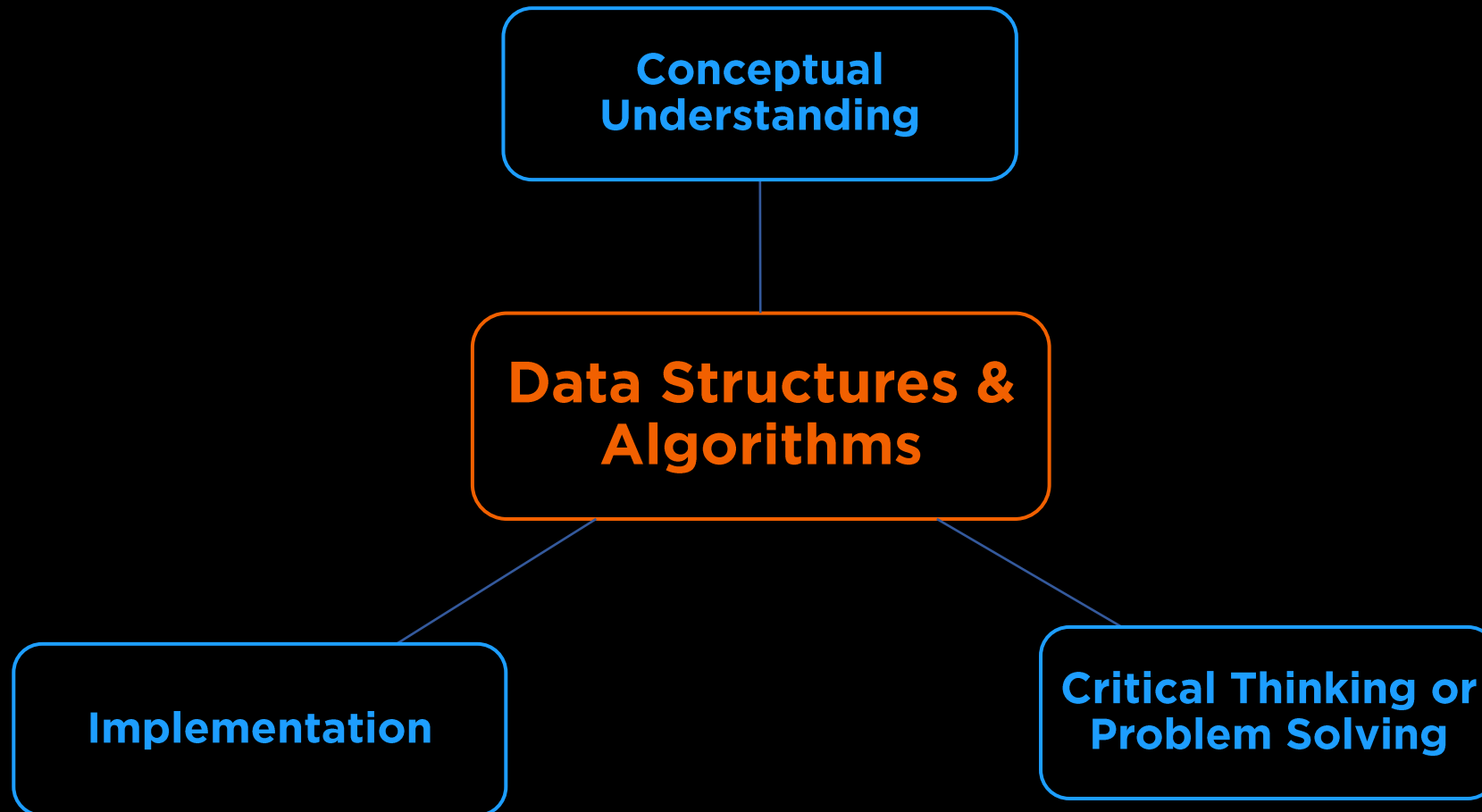
Course Objectives

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

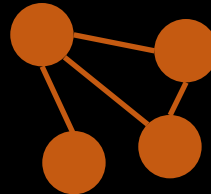
Queues



Non-linear Ordered

Trees

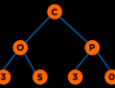
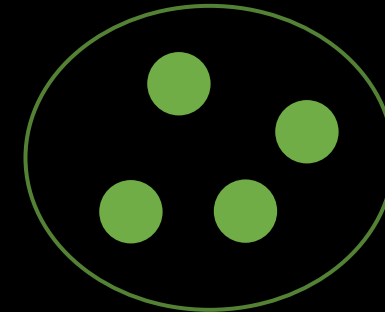
Graphs



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

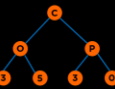
Minimum Spanning Tree

Shortest Paths

Dynamic Programming

Knapsack

Fibonacci



Logistics & Policies

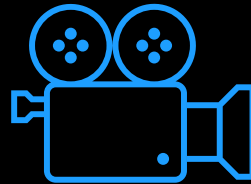
Format



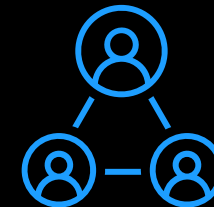
Action Items by
Monday morning



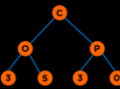
Attend discussions on
Tue.



Watch lectures **synchronously** on MWF
via Zoom. Monday's lecture will also
be in person. UFOL students watch
prerecorded videos.



1. Conceptual Quiz due on Sun.
2. Programming Quiz due on Sun.



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



- Wednesday 5-6 pm
- Friday 1-2 pm
- By Appointment (24 hours in advance)

Communication



Email

Fine for

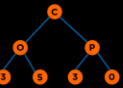
- **Personal**
- **Accommodations**
- **Emergencies**



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) | 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 | 15-May | 21-May | Course Introduction / Algorithm Analysis | Q_1 |
| 2 | 22-May | 28-May | List, Stacks, & Queues / Trees & Traversals | Q_2 |
| 3 | 29-May | 4-Jun | Trees & Traversals / Balanced Trees 1 | Q_3 |
| 4 | 5-Jun | 11-Jun | Balanced Trees 2 and Project 1 Overview | Q_4 |
| 5 | 12-Jun | 18-Jun | Balanced Trees 2 / Heaps & Priority Queues | Q_5, P_1 |
| 6 | 19-Jun | 25-Jun | Sorting / Exam 1 | E_1 |
| | 26-Jun | 2-Jul | Summer Break | |
| 7 | 3-Jul | 9-Jul | Sets, Maps, & Hashing | P_{3a}, Q_6 |
| 8 | 10-Jul | 16-Jul | Graphs 1 | Q_7 |
| 9 | 17-Jul | 23-Jul | Graphs 2 | Q_8, P_2 |
| 10 | 24-Jul | 30-Jul | Greedy Algorithms / Dynamic Programming | Q_9 |
| 11 | 31-Jul | 6-Aug | Dynamic Programming / Exam 2 | E_2 |
| 12 | 7-Aug | 8-Aug | 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

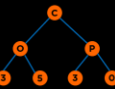
Tools

Compiler

- G++
- Stepik/EduGator/Gradescope

IDE

- OnlineGDB
- Visual Studio Code
- Clion

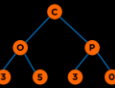


Tools

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

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 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

Request for Extensions

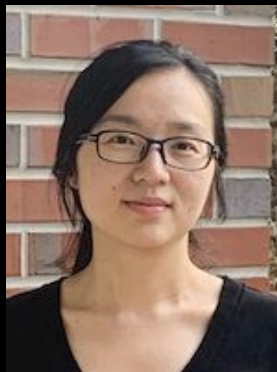
- Any request for assessment extensions should be backed by official documentation (e.g., from a medical professional, etc.) and should be sent over email. Requests without documentation will be ignored. The following reasons deem ineligibility for extensions or regrades:
 - **Failure to submit on Canvas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

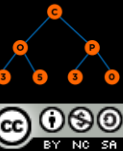
- [Dr. Josh Hug](#)
- [MIT OCW 6.006](#): Dr. Erik Demaine and Dr. Srinivas Devadas
- [Dr. Robert Sedgewick](#)

- Videos Youtube

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

- GeeksforGeeks

Walkthrough



Walkthrough

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

Questions

