Data Structures and Algorithms

Introduction to Data Structures and Algorithms

Week 1

Agenda

index	item
0	Course Overview
1	Introduction to Algorithms
2	What are Data Structures?
3	Efficiency of Algorithms
5	Next Steps

Time:
5 min

Learning Outcomes

- **Explain** various data types and algorithms, and their advantages/disadvantages
- **Analyse** when specific choices of data structures and algorithms are appropriate
- **Implement** various data structures and algorithms in Python or another language
- **Solve** interview-like problems relating to data structures and algorithms

Time:
2 min

Assessment Strategy

Assessment	**Weight**
In-class Quizzes	20%
Individual Programming Assessments	30%
Implementations	10%
Peer Projects	40%

Time:
3 min

Piazza

- Online Forum for the class - all questions should go here first
- Do not post solutions - only clarifications
- All lecture notes/slides/syllabus etc will be available here
- End of term dinner for people that contribute a lot
- Sign up - https://piazza.com/african_leadership_university/winter2020/csdsa

Time:
2 min

HackerRank

- It will be used for all programming assignments
- You will need to add comments to your code
- Automatically checks for plagiarism **so don’t cheat!!!!**
- Lots of practice problems to help you prepare. Practice when you have time!
- Sign up - www.hackerrank.com
- More problems available on app.codesignal.com - more gamified!

Time:
5 min

GitHub Classroom

- Weekly implementations should be committed to GitHub classroom repository.
- Access it here https://classroom.github.com/a/IcS_PZtR
- Create a repository called data_structures_implementations and commit it.
- Each week, create a new folder with your implementations for the week.
- Write unit tests showing that your code works for a variety of inputs
- Add comments explaining how it works and the complexity

Time:
5 min

A few things to take note:

1. Listen actively. Take notes. Attempt all algorithms on paper before you move to your PC.
2. Go through the course overview on Piazza for more info about the course
3. Do not fall behind. Playing catch up will not help you.
4. Algorithms presented will often be in pseudocode and programming implementations will be in Python. You may use other languages with my permission.
5. Practice, Practice, Practice

Time:
2 min

What is an algorithm?

Think, Pair, Share

Time:
3 min

What is an algorithm?

- A set of steps to accomplish a task.

Examples:

- Algorithm to get from home to ALU
- Algorithm to move from Mali to Madagascar
- Finding all students in this class born on a Tuesday

In this course, we will study and write some of set of steps that a computer uses to accomplish some particular tasks.

Time:
2 min

What are Data Structures?

Think, Pair, Share

Time:
3 min

What are Data Structures?

- A way of organizing and storing data in the computer memory
- As data is manipulated in memory, some operations can be performed more easily with certain data structures
- Most common operations on data structures:
    - Inputting/Inserting new data
    - Searching data
    - Deleting data

Time:
3 min

Abstract Data Types

- An abstract data type (ADT) is a theoretical or logical description
- A data structure is defined by how it is actually implemented in a particularly language.
- The same ADT can be implemented in different ways. (They are in different languages!)
- ADT = Maths, Data Structure = Machine-interpreted code.
- We will learn about abstract data types in this class and then we will implement them

Time:
2 min

How do you assess how good a data structure or algorithm is?

Time:
10 min

Asymptotic Analysis of Algorithms

- Aside correctness, we care mostly about two things when a program is running and evaluating its efficiency:
  - Space Complexity:
  - Time Complexity:
- A notation called the **Big O Notation** is used.
- An efficient algorithm consumes less of these two resources..

In-class Challenges (Groups of 1-3)

- ➔ Write a function that inputs two integers and checks if one is a multiple of another.
- ➔ Write a function that takes a string of letters and makes each letter lower case
- ➔ Write a function that takes a list and checks that every item in the list is an integer
- ➔ Write a function that takes an integer and checks if it has only even digits.
- ➔ Write a function that inputs two integers and finds their greatest common factor

Time:
7 min

Questions

Time:
3 min

Next Steps

- **Piazza** - piazza.com/african_leadership_university/winter2020/csdsa
- **HackerRank** - www.hackerrank.com (Python, Data Structures, Algorithms)
- **GitHub Classroom** - https://classroom.github.com/a/IcS_PZtR
- Start looking through the course textbook