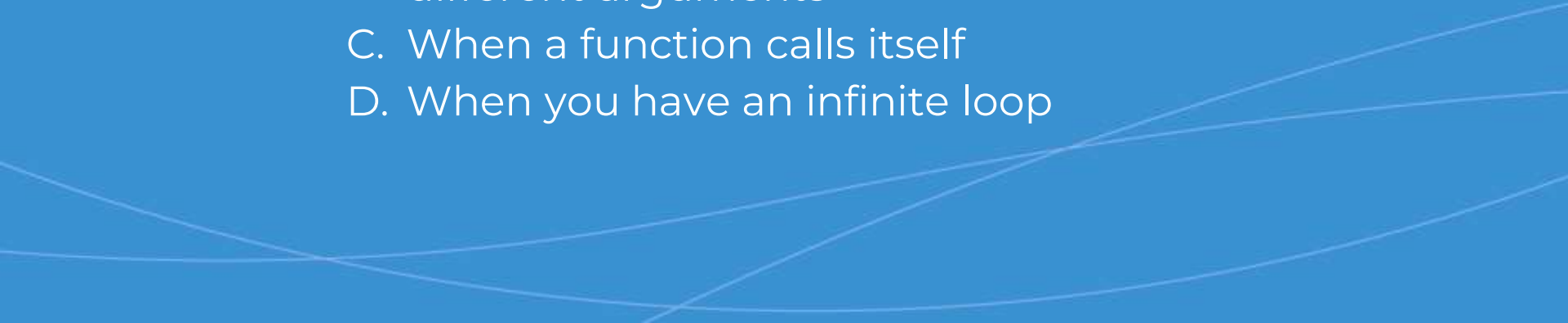# GitHub Workshop 4

Lecture Housekeeping

- The use of disrespectful language is prohibited if asking a question. This is a supportive, learning environment for all – please engage accordingly! **(FBV: Mutual Respect.)**
- No question is 'silly' – **ask away!**
- There are **Q&A sessions** midway and at the end of the session, should you wish to ask any follow-up questions. Moderators are going to be answering questions as the session progresses as well.
- If you have any questions outside of this lecture, or that are not answered during this lecture, please do submit these for upcoming Open Classes. You can submit these questions here: **Open Class Questions**

Lecture Housekeeping cont.

- For all **non-academic questions**, please submit a query:

  www.hyperiondev.com/support

- Report a **safeguarding** incident:

  www.hyperiondev.com/safeguardreporting

- We would love your **feedback** on lectures: **Feedback on Lectures**

CoGrammar

What is a recursive function

A. A function that contains a loop
B. When 2 functions have the same name but different arguments
C. When a function calls itself
D. When you have an infinite loop

What is a base case?

A. A condition that determines when we should exit a recursive function

B. A type of conditional statement for checking if a value meets one of multiple cases

C. The last condition that is checked in a conditional chain

D. Where you keep your bass guitar

Which of the following algorithms has an O(1) time complexity

A. Linear search
B. Merge Sort
C. Breadth First Search
D. None of the above

What is the time complexity of a binary search

A. O(n)
B. O(log n)
C. O(n^2)
D. O(n!)

When choosing an algorithm, what factors do we need to look at?

Recap: Recursion

★ Used to perform the same operation repeatedly
★ Provides code that is easier to read than when using iteration
★ Creating a recursive function
  ○ Define the arguments
  ○ Find the base case
  ○ Determine the steps (change in argument)
  ○ Perform the operations

Recap: Data Structures

★ We have linear data structures that are used for storing data in a directed manner
  ○ Array
  ○ List
  ○ LinkedList
  ○ Stack
  ○ Queue

★ We have non-linear data structures that can use nodes and edges to connect different data points
  ○ Graphs
  ○ Trees
★ We have key based data structures that reference values based on known keys
  ○ Hash Tables
  ○ Hash Sets

Recap: Searching

- ★ We can use algorithms to find values in our data sets
  - ○ Linear Data
    - ■ Linear Search
    - ■ Binary Search
  - ○ Non-Linear
    - ■ Breadth First Search
    - ■ Depth First Search
- ★ Linear search is the most effective for unsorted data but has a time complexity of **O(n)**
- ★ Binary search is the most efficient algorithm with a time complexity of **O(log n)**, but can only be used on sorted data

Recap: Sorting

★ We need to make use of sorted data in a lot of applications as it makes performing operations easier
★ There are hundreds of different algorithms that we can implement
★ Common Algorithms
  ○ Insertion Sort
  ○ Bubble Sort
  ○ Quick Sort
  ○ Merge Sort

Possible Words

- **Background:** . A local school is looking for a way to help their students practice their English skills.

  They have pitched the idea for a game that will provide the player with a list of letters and from that list of letters, the player will need to create a word with all of the letters that they have been given.

- **Challenge:** The school has asked you to create this game, but after running out of ideas, they have given you full creative freedom to take the game in the direction that you see best fitting.

  But there are a few hard requirements

Possible Words

Create a game that lets users create a word from a list of characters

1. **User Interaction**
   a. A list of letters should be displayed to the user
   b. The user should be able to enter their word
2. **Functionality:**
   a. The users input should include all of the letters that they were provided with
   b. When generating the characters, the application should only provide letters that will produce valid words
   c. There should be some indication whether the users input was correct.

CoGrammar

Possible Words

Create a game that lets users create a word from a list of characters

1. **Game Loop:**
   a. After each successful input, the user will be given a new set of inputs to enter
2. **Scoring System:**
   a. The user will get a point for each correct answer that they provide
   b. The scoring system can take into account the total number of letters provided
3. **Leveling System:**
   a. As the user gets more correct answers, the length of the list of letters grows.
   b. You can have different ranges lengths for each level

Possible Words

Create a game that lets users create a word from a list of characters

1. **Framework:**
   a. Make use of a module like PyGames to create a GUI, or maybe even turn the game into a web app using Django or Flask
2. **Timed:**
   a. Make the responses timed, each correct value will increase the time that the user will have on the next question.

Things to consider

1. **Application Design**
2. **Efficiency of operations**
3. **Code Structure**
4. **Edge Cases**

Application Design

You will need to think well about how your application is going to work, there are different aspects of the application that need to be taken care of.

1. User interactions
   a. How will the user enter their inputs
   b. How will the outputs be displayed
2. Generating the lists of letters
   a. What's required to generate random lists letters
   b. How to validate that the lists are random letters can make an acceptable word
3. Checking if the users input is
   a. How to make sure that the users input is a valid word.

Efficiency

The data set containing words has 10000 values, we need to make sure that the operations we perform use this large set of values as efficiently as possible.

1. Optimization
    a. Which operations need to be really fast
    b. Which operations can we make slow in order to allow the other operations to be faster
2. Storage
    a. Which data structure would be the best for storing all of the values
    b. Is there a way that we can modify the structure of the data to make the operations more efficient

Code Structure

Having a well structured code base is very important for building an application that is easy to build and manage.

1. Are there any operations that would be reused
   a. Would the application benefit from using functions
   b. Would the application benefit from using classes
   c. Would the application benefit from using modules
2. File Structure
   a. Would having the code in different files make the application easier to build (the answer is yes)

Edge Cases

We can't trust that a user will always enter the right values, so we need to guard against an possible issues.

1. What will happen if the user enters a letter of character that is not in their list
   a. Will we let the searching operation handle this
   b. Will we guard against input that will 100% be invalid

Skills you can use

OOP

★ You will need to use your knowledge of OOP to reuse code and make your code easier to work with

Data Structures and Algorithms

★ You will need to use your knowledge of time complexity as well as data structures and algorithms to build something that is efficient.

Out of the both Thinking

★ How you you use the skills you have and the data structures you know to make the application more efficient.

Questions and Answers

Questions around the Case Study