GCIS-123 Final Exam Study Guide

This is a review for the GCIS-123 final exam. This study guide can help you prepare for the final exam. Do your best to complete as many of these questions as you can.

The sections are as follows:

- **Unit 1: Computer Literacy**
- **Unit 2: Intro to Python**
- **Unit 3: Incremental Development and Testing**
- **Unit 4: Strings and Loops**
- **Unit 5: Files and Exception Handling**
- **Unit 6: Arrays, Searching, & Sorting**
- **Unit 7: Sorting**
- **Unit 8: Python Lists**
- **Unit 9: Dictionaries and Sets**
- **Unit 10: Classes & Objects**
- **Unit 11: Methods & Equality**
- **Unit 12: Stacks & Queues**

Feel free to work with each other to complete this guide. This can be great reference for studying for the final exam, so be sure to take a look at this document when you are studying for the exam.

If you think of anything else that you believe will help you study for the exam, feel free to fill in the "other" section at the bottom of the document.

You can find all of the coding activities from previous SI Sessions on this page: https://github.com/zmb6893/GCIS-123-16-Fall-SI-Activities

Activity point breakdown:

#. 1 point	**Note that all written answers are 1 pt**
#. 3 points	
#. 5 points	
#. 10 points	
#. 15 points	

Unit 1: Computer Literacy

Question	Answer
What structure are files organized in?	
Does each file have a unique absolute path?	
What command can you use to open or create a new file called “hello.txt”?	
What command is used to move into a directory?	
How can you find a txt file that starts with the letter “a”? What is this called?	
What is the workflow of git?	
What git command is used to copy the contents of a remote repository to a local repository?	
What do you call the github copy of a repository?	
Why do we use version control?	
How do you check what files have been staged to be pushed? When should you use the command?	
How do you get changes from a previous version of your program?	
What is a detached head?	
What are environment variables?	

Unit 1 Activities:

1. Clone the repository from https://github.com/zmb6893/GCIS-123-16-Fall-SI-Activities. For extra practice, you can practice activities from any of these repositories **THIS IS NOT REQUIRED!!!**.
2. Create a repository called `GCIS-123-final-review`. Navigate to that directory.
3. Create a file called `unit01.txt`
4. List **all** of the files that have a name that starts with the letter ***a***. On the first line in your `unit01.txt`, type in the title *"List of Files Starting with the Letter a"*. Copy and paste these contents on the next line.
5. Check your git log and paste the contents under a title "Git Log". Do the same with a git status.
6. Create a new directory called `Files-and-Git`
7. Move `unit01.txt` to the `Files-and-Git` directory.
8. Rename the `unit01.txt` to `responses.txt`
9. Use the git workflow while completing the rest of the activities.

Unit 2: Intro to Python

Question	Answer
How do you define a python string?	
What are invalid python variable names?	
What do you use to send information to standard output?	
What does DRY stand for?	
How do you define a python function?	
How does python handle typing?	
Where do you execute your code?	
What is a literal?	
What is the difference between a parameter and an argument?	
What is scope? How do you determine scope?	
What notation is used to access functions from an outside file or library?	

Unit 3: Incremental Development and Testing

Question	Answer
What is incremental development?	
What is a helper function?	
How often should you commit?	
Why do we do incremental development?	
How do you send information out of a function?	
As an interpreted language, when is python syntax verified?	
What are the three main types of errors?	
What is the most recent function call in the stack trace?	
What is a variable with a value that doesn't ever change?	
What is TDD?	
What are the three steps of TDD?	
What is stubbing?	
What are two qualities a unit test should have?	
What are three components of a test?	

Unit 4: Strings and Loops

Question	Answer
What is a boolean?	
Is else if valid syntax in Python?	
What is used to tell how much code has been tested?	
When does the debugger stop running code?	
What is an iteration?	
What is an escape sequence used for?	
What does len() do?	
A_string = “no way”. What character is A_string[2]?	
While loops can break your computer and your mind when what happens?	
Why would you use a while-loop over a for-loop?	
What are some good ways to refactor your code?	
Can you use a for loop to iterate over characters in a string? how?	

Unit 5: Files and Exception Handling

Question	Answer
How do you make a file accessible in your python code?	
Can a file be seen as a sequence of lines in a for loop?	
Why should you close a file after you're done modifying it in your program?	
What can you do to open and close a file in one line of code?	
What function eliminates white space?	
Do runtime errors cause exceptions to be raised? Semantic errors? Syntax errors?	
How would you create an exception?	
Can you catch specific exceptions? All exceptions?	
What happens when you don’t handle an error that has been raised?	
Why would you want to reraise an error?	
How do you read from a csv file in python?	
Why would we want to use a csv reader over split?	
What the heck is a regular expression?	

Unit 2-4 Activities:

1. In your `GCIS-123-final-review` repository, make a directory called `Flowers`.
2. Make a file called `flower_pictures.py`. Stub out a function called `draw_petal(size, color)`. Stub out a function called `draw_flower(size, layers, color)`. Stup out a function called `draw_and_print_flower(size, layers)`. Keep in mind that `draw_petal(size)` will return the coordinates for the tip of the petal as a tuple. `draw_flower(size, layers, color)` will return the coordinates of the tip of each petal as a list of tuples. `draw_and_print_flower(size, layers, color)` will return a string with the following format in the terminal:

```
A red flower of size 30 is being drawn with 3 layers…

    Location of each tip:
        (0,30)
        (16,16)
        (0,-30)
        . . .
```

3. Make a file called `test_turtle_activities.py`.
4. Make a tests for the following scenarios:
    a. Calling `draw_petal(10)`
    b. Calling `draw_petal(50)`
    c. Calling `draw_petal(100)`
    d. Calling `draw_flower(30, 3)`
    e. Calling `draw_flower(30, 5)`
    f. Calling `draw_and_print_flower(10, 3, "red")`
    g. Calling `draw_and_print_flower(30, 1, "blue")`
5. Make sure that your test is failing when you run it.
6. Implement `draw_petal(size)`
7. Run your tests and note which ones are passing.
8. Implement `draw_flower(size, layers)`
9. Run your tests and note which ones are passing.
10. If all your tests are passing, call `draw_and_print_flower(30, 3, "red")` from `main`.

Unit 6: Arrays, Searching, & Sorting

Question	Answer
What is an array?	
Can you iterate over an array? Why or why not.	
What is a search algorithm?	
What is time complexity?	
What is the best, worst, and average time complexity of linear search?	
What is recursion?	
Why would we use recursion in some cases?	
What happens if you don't have a base case in a recursive function?	
What is a stack frame?	
What is the best, average, and worst time complexity of binary search?	
What is the precondition for binary search?	
What is more efficient: logarithmic time or linear time?	

Unit 7: Sorting

Question	Answer
Can you pass a function name in as a parameter?	
What is sorting?	
What is the best, average, and worst time complexity for an insertion sort?	
What is a comparator?	
What type of algorithm is merge sort?	
What is the time complexity of split?	
What is the time complexity of merge?	
Why is merge sort an O(nlogn) algorithm?	
Is merge sort more efficient than iterative and bubble sort?	
What is the pivot in quicksort?	
How do you return and receive multiple values in python?	
What is the best, average, and worst time complexity of quicksort?	
Is the pivot always at the center of the data set?	

Unit 8: Python Lists

Question	Answer
What does it mean to be immutable? What are some immutable data types in python?	
What are some mutable data types in python?	
How do you make a tuple?	
What is a reference type?	
How does list concatenation work?	
What is the average time complexity of popping and inserting into a list?	
Difference between deep and shallow equality>	
What is slicing?	
How do you slice?	
How do you specify order with .sort()?	
How do you make a key?	
What makes a list and tuple different?	
What is list comprehension?	
How do you do list comprehension?	

Unit 9: Dictionaries and Sets

Question	Answer
Why are sets unique?	
What is the keyword to check if there is an item in a set?	
What is the time complexity of accessing an element in a set/dictionary?	
What is union?	
What is an intersection?	
What are two ways to make a dictionary?	
Are dictionary elements unique? What about the keys?	
Are sets iterable? What about dictionaries?	
How do you get the ASCII value from a string? What about the reverse?	
Do sets or dictionaries use hashing?	
What is a collision?	
What are three characteristics of a good hash?	
What is the time complexity of resolving collisions?	
What is open addressing? Chaining?	

Unit 10: Classes & Objects

Question	Answer
What is the difference between a class and an object?	
What is it called when you put information into a class?	
What notation do you use to access a field from a class?	
Why do we use encapsulation?	
Are you required to pass every slot item into the constructor?	
What is self?	
Why do we use slots?	
Why use self over static attributes?	
What is state?	
Why would we use Object Oriented Programming?	
Why do we use noun-verb analysis?	
What do verbs represent in noun-verb analysis?	
What do nouns represent in a noun-verb analysis?	

Unit 11: Methods & Equality

Question	Answer
What is the difference between a method and a function?	
Should methods always do something with the fields of a class?	
What are private fields in python? How do you make a private field?	
How do you allow other classes to interact with private fields?	
Should you make accessors and mutators for all private fields in a class?	
What is a special method?	
List 3 special methods with their parameters that are used for comparing objects.	
What is the special method that is used when you print and object?	
What is a downside to using pythons ability to dynamically add fields?	

Unit 6-11 Activities:

1. In your `Flowers` directory, create a new file called `flowers.py`. Import your `flower_pictures.py` file.
2. Create a class for a `Flower` that has a name, color, size, and number of layers. Add a constructor and accessors.
3. Make the flower hashable. What is an appropriate hash for a flower?
4. Add a method called `preview_flower(flower)`. This should show a preview of what the flower looks like using a function you made in the **Unit 2-4 Activities**
5. Create a class for a `FlowerShop`, which will have a name, catalog, and rating. The catalog will have a list of prices associated with a flower. What data structure will you use for your catalog? **Check your answer with your SI Leader.**
6. Create flowers from the `flowers.csv`. Handle errors appropriately. Store the flowers in an appropriate data-structure. **Check your selected data-structure with your SI Leader.**
7. Create flower shops from the `flower_shops.csv`. Handle errors accordingly. How will you store these?
8. Create prices for each shop catalog using random numbers ranging from 1-10.
9. Add a `__str__` method to both classes with appropriate formatting.
10. Allow the user to enter in a flower name to standard input. Display all the flower shops that hold that flower in order of ascending prices. The user should be prompted for input until they type in x, at which point the program exits.

```
Hello, would you like to browse our shops(B) or search for an
item?
     S
Type in the flower you would like to view:
     Daiisy
Sorry, none of our shops carry that; try entering something
else:
     Daisy
The following flower shops carry a daisy. You may choose to
view by price(P), value(V), or rating(R).

     Antique Flowers       5 stars: Daisy-$1.50

     Blooming Boutique     3 stars: Daisy-$1.50
     S
Here are your options!
```

Return value from the catalog in sorted order

Unit 12: Stacks & Queues

Question	Answer
What order is a stack?	
Is a stack an iterative or recursive data structure?	
What data structure is the basis for a stack?	
What are the components of a node and how are they used in a stack?	
How do you know if a stack is empty?	
What type of data structure is a queue?	
Where are new values added during an enqueue?	
What happens if the number of elements in your queue exceeds the size of the queue's array?	
How does enqueue work on a node based stack.	
What is an abstract data-type?	
What are the main two methods of a queue?	
What are the main two methods of a stack?	
What is the downside of using a node based list?	
List some abstract data types.	

Other (Coming up with a good question is worth 1 pts. The answer is worth 3 pts):

Question	Answer