

DURHAM
COLLEGE

COSC 1100



Class Exercise



COSC 1100

Class Exercise



Functions: A Smarter Way to Design Algorithms
Part 1 of 2



Functions: A Smarter Way to Design Algorithms

Goal of Part 1:
Experience designing with Functions.

Requirements

- Did you come prepared?
 - Did you view the video and other content about Functions?
 - Hopefully you are comfortable with everything up to this point!
- Collaboration.
 - You can work alone or with 1-2 colleagues during this exercise. (i.e. groups of 1-3)
 - The group will share their planning ideas at the end of this class, which will contribute to your grade.

Let's Start by Taking Up Last Week's Work in Full

We did most of it, but let's look at the rest, and review.

Let's Name Some Functions You Already Know!

- `print()`
-
-
-
-
-

Keywords: Function and Method

- A procedure or codes that can be called anywhere in a program is called a *function*
- When a function is associated with a data object and therefore is only usable in respect to that object, it is called a *method*

```
# A function you know  
print("Hello, World!")  
  
# A method of the List class.  
os_list.append('Windows')
```

Functions That Are Handy with Lists and Tuples

Function	Description
<code>len()</code>	Returns the number of elements in the list or tuple
<code>list()</code>	Converts something – like a tuple – to a list
<code>max()</code>	Returns the highest value item in the list or tuple
<code>min()</code>	Returns the lowest value item in the list or tuple

Methods and Functions for Lists and Tuples

- There are lots of functions that work on both lists and tuples
- Remember, lists can be changed and tuples are immutable
- There are many methods of the list class that don't exist for tuples!

```
# Declare identical list and tuple.  
my_list = [1, 2, 3]  
my_tuple = (1, 2, 3)  
  
print(min(my_list))    # Prints the smallest value ie. "1"  
print(min(my_tuple))  # Prints the smallest value ie. "1"  
  
my_list.append(4)       # Adds 4 to the list.  
my_tuple.append(4)      # Crash; this does not exist.
```

Methods That Are Handy with Lists (but not Tuples)

Function	Description
<code>append()</code>	Adds an element to the end of a list
<code>clear()</code>	Removes all elements from a list
<code>index()</code>	Returns the index of a specified value
<code>pop()</code>	Removes the element at a specified index
<code>remove()</code>	Removes an element with a specific value
<code>sort()</code>	Attempts to sort the contents of the list

Defining a Function

```
# Function hello_world
def hello_world(name:str = "Optimus Prime",
                course:str = "Cybertron"):
    # print Hello, World
    print("Hello, World!")
    # print name, Welcome to Course!
    print(name + ", Welcome to " + course + "!")

# Calling function hello_world with keyword and a string
hello_world()
hello_world("Samson")
hello_world("Samson", "Intro to Programming")
hello_world(course = "Intro to Programming")
```

Making Up Functions for Common List Methods

- `list.reverse()`
- `list.min()`
- `list.max()`
- `list.sum()`
- What about `sort()` ?

The Problem, Starring You! (Function Mode!)

Consider the way we have calculated your grade when we discussed arrays in-class. It was probably obvious (or maybe openly discussed with the class) that there was some redundancy. For details on this calculation, you can refer back to the course outline for details or your work on the Class Exercise for arrays.

The task for this week is to reduce the repetitive parts of your program code with functions. Let's see how much smaller you can make your program! You can probably save at least 30 lines of code.

If you prefer, you could instead try and look at the problem from Assignment 1 and consider how you could improve/streamline that problem using functions instead.

Re-Plan and Desk Check your Program!

1. Alter your design from your previous work in such a way that it uses functions. At minimum, you should consider:
 - One function to simplify the input; and
 - A second function as part of the process. Probably.
2. Consider, how could you desk check a function?
Try to produce a desk-check for each function.
3. Submit your group's finished/revised plan and desk-check(s) to DC Connect (**COSC 1100 -> Activities -> Assignments -> Class Exercise: Functions**) or MS Teams, including all group members' names.

This is meant to be done during class time; submissions received after our class time ends may be considered late.

COSC 1100

Class Exercise



Functions: A Smarter Way to Design Algorithms
Part 2 of 2



Functions:

A Smarter Way to Design Algorithms

Goal of Part 2:

Experience writing a program with functions.

Requirements

- Did you come prepared?
 - Did you view the video and other content about Functions?
 - Hopefully you are comfortable with everything up to this point!
- Collaboration.
 - Once again, while you've got access to plans and you can feel free to discuss, *everyone will do their own coding today!*
 - Each student must submit their finished application to earn full marks on this week's Class Exercises.

The Problem, starring you! (Function Mode!)

Consider the way we have calculated your grade when we discussed arrays in-class. It was probably obvious (or maybe openly discussed with the class) that there was some redundancy. For details on this calculation, you can refer back to the course outline for details or your work on the Class Exercise for arrays.

The task for this week is to reduce the repetitive parts of your program code with functions. Let's see how much smaller you can make your program! You can probably save at least 30 lines of code.

If you prefer, you could instead try and look at the problem from Assignment 1 and consider how you could improve/streamline that problem using functions instead.

Plan to Program!

- With consideration to the plan you have developed earlier, either edit or re-code the application using Python!
 1. Code your solution to the given problem based on your own planning notes.
 2. Chat, ask lots of questions, and don't be afraid of mistakes.
 3. Have it checked during class, and/or submit your finished Python code file to DC Connect
(**COSC 1100 -> Activities -> Assignments -> Class Exercise: Functions**).

This is meant to be done during class time; submissions received after our class time ends may be considered late.

Class Exercise Part 2 Summary

- What other uses of Functions can you think of?
- Should desk checking just the functions before using the function in your code give you more confidence in using the function? Why? Why not?
- Can a function call itself?