Midterm Topics:

Basic Programming Mechanics

- Functions
  - What is a function?
  - How to call a function
  - Arguments
  - Return Values
- Commenting your code
- Variables
  - What is a variable?
  - Creating variables
  - Using variables in expressions
  - Naming rules
- Reading input from the keyboard with the input() function

Math Expressions

- Math operators (+, -, /, //, *)
- Writing math expressions
- Evaluating math expressions
- Storing & printing the results of math expressions
- Difference between the two division operators (/ and //)
- Order of operations in math expressions
- The exponent operator (**)
- The modulo operator (%)

Data Types

- What is a data type?
- Strings
- Numeric data types
  - Integers (int)
  - Floating point numbers (float)
- Mixed type expressions
- Data type conversion
  - Using the float() and int() function to convert strings into numbers
  - User input & data types (converting strings to floats / ints for calculation purposes)
- The Boolean data type
- Boolean variables

Output with the print() function

- General use of the print function and its default behavior
  - Unlimited arguments
  - Spaces inserted between arguments
  - Line break after each call to the function
- Customizing line endings (end=”)
- Customizing argument separators (sep=”)
- Escape characters (\n, \t, etc.)

Basic String Manipulation

- Combining two strings (concatenation) – "+" operator
- Multiplying a string (repetition) – "*" operator
- Formatting numbers using the format() function
  - Formatting Strings – width, left align, right align, center align
  - Formatting Integers – width, left align, right align, center align
  - Formatting Floats – width, left align, right align, center align, # of decimal places, "," separator
- Case manipulation using str.lower() and str.upper()
- Calculating string length using the len() function

Selection Statements

- The structure of an IF statement (IF keyword, condition, colon, indentation)
- Writing a condition for an IF statement
- Boolean operators (<, >, ==, !=, >=, <=)
- Comparing numeric values using Boolean expressions
- Comparing string values using Boolean expressions
- Using the IF-ELSE statement
- Nesting decision structures (IF statements inside other IF statements)
- The IF-ELIF-ELSE statement
- Logical operators (and, or, not)

Condition Controlled Loops

- The structure of a "while" loop
- Mechanics & how they work
- Setting up conditions for a while loop
- Infinite loops and how to work with them
- Sentinels (defining a value that the user enters that causes the loop to end)
- Input validation loops (asking the user to continually enter a value until that value matches some condition)
- Setting up and using accumulator variables
- Self referential assignment statements (i.e. counter = counter + 1)
- Augmented assignment operators (i.e. counter += 1)

The Range Function

- mechanics and how the function works
- creating simple ranges (i.e. range(5))
- creating ranges with defined start and end points (i.e. range(3,10))
- creating ranges with a step value (i.e. range(5,50,5))
- creating ranges that count backwards (i.e. range(50,5,-5))
- user controlled ranges (i.e. range(1, somevariable))

Functions

- mechanics and how functions work
- function definitions
- arguments
- return values
- calling a function
- local variables (variables that are defined inside a function and can only be accessed inside that function)
- passing arguments to your own functions
- passing multiple arguments to your own functions
- global variables (variables created outside a function that can be accessed by any part of your program)
- making changes to global variables inside a function using the ‘global’ keyword
- writing a value returning function (i.e. using the ‘return’ keyword to send a result from your function to the part of your program that called that function)
- returning multiple values from a function
- Input, Processing & Output notation

Miscellaneous Concepts

- Generating random numbers
- Errors & error types
- Debugging strategies
- Pseudocoding