Prior Exam Questions:

1. Write a Python program that takes as inputs 5 integers. The program should check to see if any of the 5 are duplicates of another (i.e., check whether any of the integers were entered more than once). If, after all inputs are entered, a duplicate is found, the program should print “Duplicates”, otherwise it should print “All Unique”.

2. Write a program that will ask a user to enter names and ages of people, stopping when an age of 0 is entered (and not processing that person). The program should collect this information, and then output the average age, the name of the oldest person, and the name of the youngest person. Assume no two people have the same age.

3. Given a list of words stored in the variable List_words, output the longest word and the length of the longest word (assume that there will be only 1 word of the longest length). Output may look like:

```
The longest word ‘champions’ has 9 characters.
```

4. The Maclaurin series expansion for 1/(1-*x*) on an interval from $-1 < x < 1$ is as follows:

$$\frac{1}{1-x} = \sum_{n=0}^{\infty} x^n = 1 + x + x^2 + x^3 + x^4 + \cdots + x^n$$

Write Python code which asks for input of a value of *x* on the interval $-1 < x < 1$, and which computes an approximation to 1/(1-*x*) using the using the series expansion summation. The summation should be continued until the term to be added to the summation is less than 10^{-6} in absolute value. Hint: Note that each term in the series is x raised to a power, including the 1 and x terms: $x^0=1$ and $x^1=x$

5. Explain when one would use a while loop, and when one would use a for loop.

6. In your own words, explain what is meant by the Pyramid approach to program development.

7. For the Python code given below, what would be printed to the screen? If the code would produce an error, write “Error”. If there would be no output, write “None”.

```
x = 10
y = 5
if x%2 == 0:
       if y>5:
       print("A")
       else:
       print("B")
       print("C")
else:
       if y<5:
       print("D")
       else:
       print("E")
       print("F")
print("G")
print("H")
```

8. For the Python code, what would the outputs be? Write them in the blanks spaces provided. Only fill in the blanks when there actually is output!

```
x = 2
y = "A"
while x<100:
        print(x, y)
        x *= x
        y += y
```

9. Write a Python program which uses loops that can output the following pattern.

```
1
22
333
4444
55555
666666
7777777
88888888
```

10. Given a list xdata of unknown length that contains x values, write the code to calculate a y-value for each x-value using the equation below. Store the calculated y values in a list called y_data. Do not use numpy arrays for this problem. Please use Python list functions, list methods, and list operators.

$$y = 4.12x^2 + 1.52x - 7.1$$

11. List 4 different data types that we have encountered so far in this course (excluding lists, arrays, tuples, complex variables) and then assign one example value of each type to the variables listed below

12. How does the % operator differ from the / operator in Python?

13. Write the Python code that will estimate the derivative of the following function at a point x . You will need to input a value of x, then check to ensure that only values of x>0 can be used in the calculation, and output the answer.

$$f(x) = 4^x - 5\log_9(x)$$

Write the Python code that will estimate the derivative of the following function at a point x . You will need to input a value of x, then check to ensure that only values of x>0 can be used in the calculation, and output the answer.

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

In Python, the function to take the log of a number to base ***a*** is log(***x***,***a***) where ***x*** is the number and ***a*** is the base. (In this problem ***a***=***9***). The log`()` function is available in the *math* module and will only work for ***x***>0. The output from the code should be:

```
The derivative for the function at x = 0.6  is  -0.6078.
```

14. Sheldon Cooper's (of Big Bang Theory) favorite number is 73. One of his reasons is that 73 is a prime number and there are 21 prime numbers between 1 and 73. A prime number is a number greater than 1 that is not divisible by another number (the only even number that is a prime number is 2; all other prime numbers are odd). Write the Python code to calculate and print the prime numbers between 1 and 73 (but not including 73). Your program will also need to count the prime numbers to see if this is really the 21th prime number.

15. The series expansion for ln $\left(\frac{1+x}{1-x}\right)$ with -1 < x < 1 is expressed as

$$\ln\left(\frac{1+x}{1-x}\right)=\sum_{n=1}^{\infty}\frac{2}{(2n-1)}x^{2n-1} \quad \text{or} \quad \ln\left(\frac{1+x}{1-x}\right)=2x+\frac{2}{3}x^{3}+\frac{2}{5}x^{5}+\frac{2}{7}x^{7}+\ldots$$

Write the Python code to input a value of x from the interval $-1 < x < 1$, verify that x is within this range (if not re-enter a value within the range), and then evaluate the series expansion summation for ln $\left(\frac{1+x}{1-x}\right)$ for the value of x until the absolute value of a term in the summation is less than the TOL, which is 1.0E-06. For example, if x = 0.5, the first term is for n=1 is $\frac{2}{(2*1-1)}x^{2*1-1}$, or 2$x$. Since this term is > TOL, add the term to the summation variable and continue. This logic continues until one of the terms will be < TOL, and the summation stops and outputs the results. The natural log function in Python can be found in the math module with the format `math.log()`.

16. Without using a list, array, or tuple, write the Python code that will input a series of positive floating point numbers until a negative value is entered. The program should then output the maximum number, the minimum number and the average number for this series.

17. What is the output of the code shown here?

```
x = 3
y = 4
if x%2 == 0:
        if y>3:
        print("A")
        else:
        print("B")
        print("C")
else:
        if y<5:
        print("D")
        else:
        print("E")
        print("F")
print("G")
```

18. What is the output of the code shown here?

```
for i in range(12):
        if i%2!=1 and i%3==0:
                print(i)
```

19. What is the output of the code shown here?

```
x = 2
y = "A"
while x<100:
        print(x, y)
        x *= x
        y += y
print(x)
print(y)
```

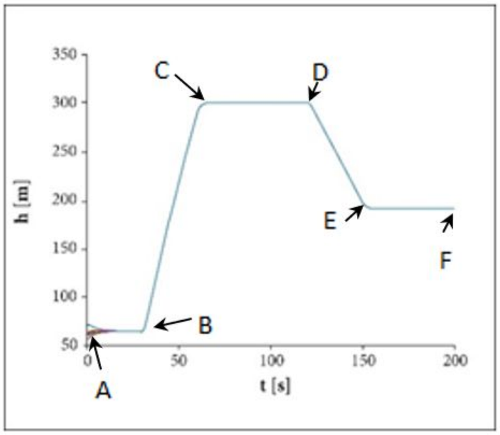
20. If we have a list called L of length > 10, using list slicing create a sub list called L_new of the 4th through the 7th elements of L. Now show how to modify the list L by removing the 2nd, 3rd, and 4th elements.

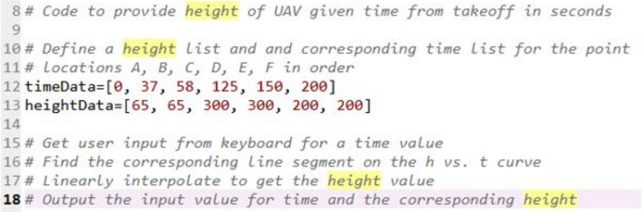
21. Given x=3 and y=5
Evaluate the following Boolean expressions:

(a) x != y – 2
(b) x >= 0 and not x < 10
(c) x < 0 and x < 10
(d) x >= 0 and x < 2
(e) x < 0 or y < 5
(f) not x > 0 or x < 10

22. The figure below depicts height above the ground (h) in meters versus time (t) in seconds. The data corresponds to a test flight involving several unmanned aerial vehicles (UAVs). We have a need for a Python code to predict the height above ground given time from the beginning of the flight. Your co-worker started the code as seen below, but you've been asked to finish it. Assume that height varies linearly with time in between the points shown. Your code should perform the following tasks:

- Print a message to the screen describing what the code does.
- Get input from the keyboard representing the time. (Don't forget the prompt.)
- Check that the time entered is valid, and warn the user if invalid input.
- Perform linear interpolation on the section of the curve that corresponds to the time value input to get the corresponding height value.
- Print a nicely-formatted result to the screen that includes both the time entered and the corresponding height.



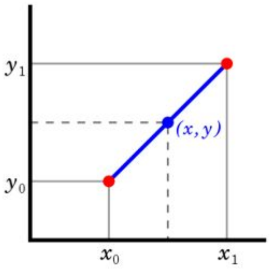


```
# Code to provide height of UAV given time from takeoff in seconds

# Define a height list and and corresponding time list for the point
# locations A, B, C, D, E, F in order
timeData=[0, 37, 58, 125, 150, 200]
heightData=[65, 65, 300, 300, 200, 200]

# Get user input from keyboard for a time value
# Find the corresponding line segment on the h vs. t curve
# Linearly interpolate to get the height value
# Output the input value for time and the corresponding height
```

Interpolation is the estimation of a value between two known points. Given the graph below, with known points (***x_0, y_0***) and (***x_1,y_1***), you would use these points to calculate a value of ***y*** for a given ***x*** value.


y_1
(x, y)
y_0
x_0
x_1