Mission 9 - Line Following Review Questions

Select the computer science definition of: GLOBAL	A. A container that holds keys and values. B. A variable defined outside a function. C. A variable created inside a function. D. A read-only list.
Select the computer science definition of: LOCAL	A. A container that holds keys and values. B. A variable defined outside a function. C. A variable created inside a function. D. A read-only list.
Select the computer science definition of: TUPLE	A. A container that holds keys and values. B. A variable defined outside a function. C. A variable created inside a function. D. A read-only list.
Select the computer science definition of: DICTIONARY	A. A container that holds keys and values. B. A variable defined outside a function. C. A variable created inside a function. D. A read-only list.
REPL can be used for all the following EXCEPT:	A. Default parameters B. Output messages C. Input strings D. As a calculator
What is the correct code for list comprehension?	A. list=[for i in range(5) ls.read(i)] B. list=(for i in range[5] ls.read(i)) C. list=[ls.read(i) > thresh for i in range(5)] D. [ls.read(i) > thresh for i in range(list)]
What data type does ls.check() return?	A. List B. Tuple C. String D. Boolean
When working with a tuple, which of the following will cause an error? `my_tuple = (True, True, False, False, True)`	A. number = len(my_tuple) B. result = my_tuple[0] C. leds.ls(my_tuple) D. my_tuple.append(True)
Which of the statements with logical operators is True?	A. True and False B. True or False C. False or False D. False and True

Given the code, what will print:
`number = 5`
`status = False`
`if number < 1 or status:`
`  print("Good to go")`
`elif not status:`
`  print("On hold")`
`else:`
`  print("Abort")`
	A. Good to go
	B. On hold
	C. Abort
	D. On hold, and then Abort

Given the code, what will print:
`for count in range(5):`
`  if count < 2:`
`  print('A', end=' ')`
`  elif count < 4:`
`  print('B', end=' ')`
`  else:`
`  print('C', end=' ')`
	A. A B B C C
	B. A A B B C
	C. A A B B C C
	D. A B B B C

What is assigned to a variable to represent no value?	A. my_var = None B. my_var = Undefined C. my_var = 0 D. my_var = “none”

What does this code do?
`prev_vals = (0, 0, 0, 1, 1)`
`vals = (1, 1, 0, 0, 0)`
`if vals != prev_vals:`
`  print(vals)`
`  prev_vals = vals`
	A. Nothing, condition is false
	B. Nothing, condition is true
	C. Prints vals and updates prev_vals
	D. Updates prev_vals and then prints it

Given the code, what is the value of **amount**?
`veggies = {'carrots':10, 'beans':5, 'peas':8}`
`amount = veggies['beans']`
	A. 10
	B. 5
	C. 8
	D. KeyError

Given the code, what is the value of **amount**?
`veggies = {'carrots':10, 'beans':5, 'peas':8}`
`amount = veggies['corn']`
	A. 10
	B. 5
	C. 8
	D. KeyError

Given the code, what is the value of **amount**?
`veggies = {'carrots':10, 'beans':5, 'peas':8}`
`amount = veggies.get('corn', 10)`
	A. 10
	B. 5
	C. 0
	D. KeyError

What is the purpose of the global statement?	A. Allow the function to be called anywhere B. Allow a local variable to be used outside the function C. Allow a global variable to be updated inside a function D. Keep a global variable from being updated inside a function

Given the code, what is printed?
`name = 'Barbie'`
`def fun():`
`  name = 'Ken'`
`  print(name)`
 	
`fun()`
	A. ‘Barbie’
	B. ‘Ken’
	C. Nothing is printed
	D. UnboundLocalError

Given the code, what is printed?
`name = 'Barbie'`
`def fun():`
`    print(name)`
`    name = 'Ken'`
 	
`fun()`
	A. ‘Barbie’
	B. ‘Ken’
	C. Nothing is printed
	D. UnboundLocalError

Given the code, what is printed?
`name = 'Barbie'`
`def fun():`
`    global name`
`    print(name)`
`    name = 'Ken'`
 	
`fun()`
	A. ‘Barbie’
	B. ‘Ken’
	C. Nothing is printed
	D. UnboundLocalError