**Student Name \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Section Number \_\_\_\_\_\_\_\_\_**

*This exam is double-sided. Be sure to pay attention to all instructions.*

Q1. [**32 pts.**] Compute the value of these expressions assuming the following variable values.

vals = [2, 0, 3, 4, 2]  
other = [0, 3, 4]  
third = [2, 4, 6, 8]  
x = 3

|  |  |  |
| --- | --- | --- |
|  | Expression | Expression Value |
| a) | range(len(vals)) |  |
| b) | vals[2:4] |  |
| c) | vals.index(x,2) |  |
| d) | vals[4] == 4 |  |
| e) | other **in** vals |  |

|  |  |  |
| --- | --- | --- |
|  | After executing… | Then vals is equal to … |
| f) | vals.remove(2) |  |
|  | After executing… | Then other is equal to … |
| g) | other[0] = other[2] = other[1] |  |
|  | After executing… | Then third is equal to … |
| h) | **for** element **in** third:  element = element \* element |  |

For the following expressions, your job is to write the appropriate value **after** each of these individual statement blocks executes.

|  |
| --- |
| Q2 Sample Output |
| >>> isBalanced([1,2,2,2,4],2)  **True** >>> isBalanced ([1,2,4,8],3)  **True** >>> isBalanced ([2,3,6,7],6)  **False** |

Q2. [**16 pts.**] A list is *balanced* with respect to a *pivot* value if the number of its elements strictly smaller than pivot is equal to the number of elements strictly greater than pivot.

Write a function isBalanced() that takes two parameters, a list and a value pivot and returns True if the list is balanced, but False otherwise.

Q3. [**16 pts**.] The isArithmetic function takes two parameters, a list and an integer k. This function intends to return True if list represents an arithmetic sequence where each element (other than the first one of course) is equal to the previous element plus k. The function returns False otherwise. The following is the expected output

|  |
| --- |
| Q3 Expected Output |
| >>> isArithmetic([1,2,3,4],1)  **True** >>> isArithmetic([1,2,4,8],2)  **False** >>> isArithmetic([13, 10, 7], -3)  **True** |

You must find four defects in the code below:

def isArithmetic[list, k]:

"""Returns True if list defines arithmetic sequence by adding k'''

for idx in range(len(list)):  
 if list[idx] == list[idx-1] + k:  
 return True

Circle four defects **and explain how you would fix them**.

Q4. [**18 pts.]** In this question you are to demonstrate that you can hand-execute Python code. Below start by executing main() and then report on the value of list at three marked places.

|  |
| --- |
| **def** trial(list):  **for** idx **in** range(4):  **if** idx **in** list:  **del** list[idx]  **def** main():  **A) What is value of list?**  list = [8, 3, 6, 1, 3]  **for** idx **in** range(1,4):  list[idx] = list[idx] \* idx    # A) WHAT is value of list here?  trial(list)  # B) WHAT is value of list here?  **B) What is value of list?**  list = [5, 3, 1, 2, 4]  trial(list)  # C) WHAT is value of list here?  **C) What is value of list?** |

Q5. [**18 pts.**] You are provided with the following function (I only show you the header and its documentation):

**def** simOneGame(probA, probB):  
 """Simulate a single game of racquetball given A's and B's  
 probability of winning a server. Note that A always serves   
 first. Game is up to 15.

Returns tuple (scoreA, scoreB) with final score for A and B  
 """

|  |
| --- |
| Q5 Sample Output |
| >>> simNGames(250, .6, .7)  **(69, 181)** >>> simNGames(100, .8, .8)  **(65, 35)** >>> simNGames(100, 1.0, 1.0)  **(100, 0)** |

Using simOneGame, you are to write a function simNGames() that simulates a number of games for two racquetball players with winning serve probabilities probA and probB. simNGames shall return a tuple with the number of wins by A and the number of wins by B. Sample output for simNGames is provided above.