

Name : \_\_\_\_\_

Grading TA: \_\_\_\_\_

- **INTEGRITY:** By taking this exam, you pledge that this is your work and you have neither given nor received inappropriate help during the taking of this exam in compliance with the Academic Honor Code of Georgia Tech. Do NOT sign nor take this exam if you do not agree with the honor code.
- **DEVICES:** If your cell phone, pager, PDA, beeper, iPod, or similar item goes off during the exam, you will lose 10 points on this exam. Turn all such devices off and put them away now. You cannot have them on your desk.
- **ACADEMIC MISCONDUCT:** Academic misconduct will not be tolerated. You are to uphold the honor and integrity bestowed upon you by the Georgia Institute of Technology.
  - Keep your eyes on your own paper.
  - Do your best to prevent anyone else from seeing your work.
  - Do NOT communicate with anyone other than a proctor for ANY reason in ANY language in ANY manner.
  - Do NOT share ANYTHING during the exam. (This includes no sharing of pencils, paper, erasers).
  - If you draw a star and write your name on the last page of this exam you will receive 2 extra credit points.
  - Follow directions given by the proctor(s).
  - Stop all writing when told to stop. Failure to stop writing on this exam when told to do so is academic misconduct.
  - Do not use notes, books, calculators, etc during the exam.
- **TIME:** Don't get bogged down by any one question. If you get stuck, move on to the next problem and come back once you have completed all of the other problems. This exam has 6 questions on 8 pages including the title page. Please check to make sure all pages are included. You will have 50 minutes to complete this exam.

*I commit to uphold the ideals of honor and integrity by refusing to betray the trust bestowed upon me as a member of the Georgia Tech community. I have also read and understand the requirements outlined above.*

Signature: \_\_\_\_\_

| Question              | Points | Score |
|-----------------------|--------|-------|
| 1. Multiple Choice    | 12     |       |
| 2. Multiply String    | 3      |       |
| 3. Recursive Function | 6      |       |
| 4. Draw Me            | 10     |       |
| 5. Find Errors        | 6      |       |
| 6. Find Median        | 14     |       |
| Total:                | 51     |       |

## 1. (12 points)

For each of the following multiple choice questions, indicate the most correct answer by circling it!

- (a) [1 pt] What is pointed at by the `myList` and `sortedList` variables after the following code is executed?

```
myList = [128,100,111,31]
sortedList = myList.sort()
```

- A. `myList = [128,100,111,31]` and `sortedList = [31,100,111,128]`  
 B. Both `myList` and `sortedList = [31,100,111,128]`  
 C. `myList = None` and `sortedList = [31,100,111,128]`  
 D. `sortedList = None` and `myList = [31,100,111,128]`  
 E. `sortedList = [0,1,2,3]` and `myList = [128,100,111,31]`  
 F. `myList = [128,100,111,31]` and an exception occurs, preventing `sortedList` from being assigned.

- (b) [1 pt] What is printed when the following code is executed?

```
yourList = ['3', ['list inside yourList'], (1,2,5), 3 ]
yourList[2][2]=int(yourList[0])
print(yourList[2])
```

- A. 3    B. (1,2,3)    C. (1,2,5)    D. None    E. Nothing is printed, because an exception is thrown

- (c) [1 pt] Which of the following is definitely recursive?

- A. A function that calls itself.  
 B. A function that calls another function.  
 C. A function that contains a loop.  
 D. A function that contains a return.  
 E. Any code that fails to end if you do not include a terminating condition.  
 F. None of the above.

- (d) [1 pt] What is the value of `a` after this code executes?
- ```
a = "CS is fun"
a[2:-1]
```
- A. `'S is fu'`
  - B. `"S is fu"`
  - C. `' is f'`
  - D. `" is f"`
  - E. `'CS is fun'`
- (e) [1 pt] Assuming that `astr` has been defined as follows: `astr= "hello"` Which of the following expressions will produce an error?
- A. `astr.list()`
  - B. `len(astr)`
  - C. `astr + "goodbye"`
  - D. `astr.split()`
- (f) [1 pt] Given the following code, what is printed?
- ```
print( list( range(1, 5)) )
```
- A. `[1, 2, 3, 4, 5]`
  - B. `[1, 2, 3, 4]`
  - C. `[1, 1, 1, 1, 1]`
  - D. `[2, 3, 4, 5]`
  - E. None of the above
- (g) [1 pt] Which name below will give an error message because it is **not** a valid Python variable name?
- A. `picture-1`
  - B. `picture_1`
  - C. `picture1`
  - D. `picture`
  - E. None of the above

(h) [1 pt] Assume the following code has been executed by the python interpreter:

```
def mysteryFunc():
    x = 17
    return print(x)

def mysteryFunc2():
    for x in range(5,15):
        print(x)
    return x

x = [1,2,3]
x = x.append(4)
a = mysteryFunc()
b = mysteryFunc2()
```

Which of the following is correct?

- A. The value in x is [1, 2, 3, 4]
  - B. The value in a is 17
  - C. the value in b is 14
  - D. the value in b is 15
  - E. The value in b is 5
  - F. The value in b is None
- (i) [1 pt] In Python, the compound data type is:
- A. Integers
  - B. Floats
  - C. Booleans
  - D. Strings
  - E. NoneType
- (j) [1 pt] Immutable compound data types include:
- A. Lists
  - B. Tuples
  - C. Dictonaries
  - D. Integers
  - E. None of the above

- (k) [1 pt] The reserved word `return` in Python serves which of the following purposes?
- A. Do something again
  - B. Go back to the beginning of a loop
  - C. Retrieve a previously stored value
  - D. Interrupt the flow of execution
  - E. None of the above
- (l) [1 pt] Which of these statements is a legal expression?
- A. `"A" + "B"`
  - B. `"A" - "B"`
  - C. `"A" * "B"`
  - D. `"A" / "B"`

2. (3 points)

Pretend you are the Python interpreter and the following code has been entered and executed. Write what is printed to the screen beside the code:

```
def multiplyStr(myStr):
    newStr = ""
    for letter in myStr:
        newStr = newStr + 2*letter.lower()
    return newStr
print("Done!")

out = multiplyStr("HEY")
print(out)
```

3. (6 points)

a. Pretend you are the Python interpreter and the following code has been entered and executed. Write what is printed to the screen beside the code:

```
def thisFunc(x):
    print(x)
    if x > 1:
        result = thisFunc(x-1)
        print(result)
    return x + 1

x = thisFunc(3)
```

b. What value does the `x` variable point at after the code above has been executed?

## 4. (10 points)

Draw what appears in the window after the function below is executed. Be sure to indicate the length of each line segment, as well as the final position and orientation of the turtle. Be very careful to execute the code exactly as the python interpreter would, not as you think it should be executed.

```
import turtle

def drawMe(turtleList):
    wn = turtle.Screen()
    for item in turtleList:
        try:
            turtle.forward(item)
        except:
            turtle.backward(50)
        if item == "<-" or "l":
            turtle.left(90)
        elif item == "->" or "r":
            turtle.right(90)
    wn.mainloop()

drawMe( [100, "->", 50, "l", 100, "<-", 50] )
```

## 5. (6 points)

The following code is intended to take in a list of integers and return a new list containing all numbers from the original list that are *\*NOT\** evenly divisible by 3.

Clearly circle and EXPLAIN all errors present in the code, whether syntax-based or logic-based, that prevent this function from working as intended. Each error should be considered independent of the others (i.e. while inspecting one bit of code, you should assume the rest of the function works/runs as intended). You should assume that the input will be a valid list of only integers AND that there are multiple items in the list.

*\*Note\** There are no accidental typos in the code presented below. Any mistake that you see should be identified.

```
##Return all numbers not evenly divisible by 3
##For example, getNums([1,2,3,4,5,6]) returns [1,2,4,5]
```

```
Def getNums(aList):
    results = []
    for i in range(len[aList]):
        test = aList[i] % 3
        if test = 1 or 2:
            results.append(i)
    return results
```

6. (14 points)

Write a function called `findMedian` that accepts a single parameter, which is a list of numbers. Your function must return the median value for the list of numbers as a float. Recall that the median of a list of numbers is defined to be the “middle” number once the list is sorted. If the list has an even number of elements, you must average the two “middle” elements to find the median.

Sample runs:

```
>>> x = findMedian( [1,7,4,2] )
>>> y = findMedian( [3,0,1] )
>>> z = findMedian( [5,5,5] )
>>> print( x,y,z)
3.0 1.0 5.0
```