CS 1803 Fall 2010 Exam 1 **Name:**_____

Your Grading TA:_____ **Your Section :** _____

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 will not be tolerated. You are to uphold the honor and integrity bestowed upon you by the Georgia Institute of Technology.

a) Keep your eyes on your own paper.
b) Do your best to prevent anyone else from seeing your work.
c) Do NOT communicate with anyone other than a proctor for ANY reason in ANY language in ANY manner.
d) Do NOT share ANYTHING during the exam. (This includes no sharing of pencils, paper, erasers).
e) Follow directions given by the proctor(s).
f) Stop all writing when told to stop. Failure to stop writing on this exam when told to do so is academic misconduct.
g) Do not use notes, books, calculators, etc during the 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.

Signature: _____

Question	Possible Points	Earned Points	Graded By
1 – True / False	5		
2 – Multiple Choice	20		
3 – Code Reading	20		
4 - amountPaid	25		
5 – getRange	30		
Total	100		

Question 1: True/False - Circle the correct answer! 5 points

False – try/except statements work the same as if/else statements.

True – You can assign the result of a print function call to a variable using an assignment statement. E.g.: myVar = print("Hey!")

False – What you put in the parenthesis of a function, myFunction(**this**), is called the input.

False – If you have two if statements, such as:

```
if(x<5):
   doSomething()
if(x>5):
   doSomethingElse()
```

both function calls will always run.

False – If you have an if statement, such as:

```
if(x<5):
   doSomething()
else:
   doSomethingElse()
```

the doSomethingElse() function will always run.

Question 2: Multiple Choice / Fill in the Blank 20 points

A. What is a way to change the string nbalegend = "Larry Bird" to say "Harry Bird"

C) nbalegend = "H" + nbalegend[1:]

B. Which of the following statements about functions is true?

E) B and C

C. Which of the following is a valid line of code to open a file that you want to read?

C) f = open("filename.txt", "r")

D. Which of these does not require parenthesis?

d) return

E. The code contained in an if statement is called what?

b. Block

F. Infinite recursion is often caused by an incorrect or lack of a(n) _____:

B) Terminating condition

Questions G-J will fill in the blanks for the following code:

```
def classStanding (hours):
        status = ""
        if   __G__ <= 29:
                __H__ = "Freshman"
        __I__:
                __H__ = "Not a Freshman"
        ans = "My status is ", + status
        __J__  ans

standing = classStanding(56)
```

Choose the BEST choice to make the code valid

G. The value of space G should be

D) hours

H. The value of the spaces marked H should be

C) status

I. The value of space I should be

a) else

J. The value of space J should be

d) return

Question 3: Code Reading **20 points**

A. Look at the code below- what is the value stored in result?

```
myInput = [3,7,1,8,6]
def someFunc(funcInput):
    myVal = 5
    for item in funcInput:
          myVal = myVal + item
          x= int(len(funcInput))
          result = myVal / x
    return (result)

 result = someFunc(myInput)
```

5 points: Result is pointing at 6.0

B. Does this code run? If so, what is printed on the screen and what is the type of output?

```
def doesThisWork(string,i = 12):
                for x in string:
                      i = i + 2
                      return ( print(i) )
                      return i
output = doesThisWork("Cs 1803 is fun!")
type(output)
```

1 point: Yes, it can run
2 points: 14 (printed to the screen)
2 points (type of output is None or NoneType)

C. What is printed when this code is run?

```
def whatDoIDo(num4, num2 = 0.1):
          num1 = [1.0,"hi",'rawr']
          num3 = num1
          if(num4 == "bye" or "cya"):
             num3[0] = num2
          else:
             num1[2] = num4
          return num1

x = whatDoIDo("hello",15)
```

```
print(x)
```

5 points: [15, 'hi', 'rawr']
-1 for missing quotes.
-2 for each missing item.

D. What would the value of the retValue variable be after this code is run?

```
moreNums= [3,2.99,12.01,12,2,9]

def blank(num_list):
    variable = num_list[0]
    while type(variable) == type([]):
        variable = variable[0]
    for element in num_list:
        if type(element) == type([]):
            variable2 = blank(element)
            if variable < variable2:
                variable = variable2
                variable = float(variable)
        else:
            if variable < element:
                variable = element
                variable = int(variable)
    return variable

retValue = blank(moreNums)
```

5 points: retValue references the integer 12.

Question 4: Code Writing – amountPaid 25 points

Write a function called amountPaid()
Prompt the user to enter the pay rate and the number of hours worked.
If the user worked more than 40 hours, their overtime rate of pay is time and a half (1.5 x the normal pay rate) and only applies to the extra hours (over 40) that were worked. The function should generate and return a string telling how much the person is to be paid: "Your amount paid is: $67.50 "
You may assume that the user will only type correct floating point numbers, so no error checking is needed. The returned string must use string formatting to place the number immediately after the dolar sign, and the dollar amount must be rounded to exactly two decimal places.

Parameters: none
Return: A string stating the amount of pay

```
def amountPaid():
    r=input("Please enter your rate of pay: ")
    h=input("Please enter your hours worked: ")
    hours = float(h)
    rate= float(r)
    if(hours > 40):
        hours= hours-40
        pay = (40*rate)+(hours*(rate*1.5))
        return "Your pay is $ %.2f" %pay
    else:
        pay = hours*rate
        return "Your pay is $ %.2f" %pay
```

+2 correct function header

+3 asking user for hours worked and pay rate
+3 convert hours worked and pay rate to float
+3 conditional statement to determine if there is overtime
+3 calculate number of overtime hours (correctly)
+3 calculate overtime pay rate
+3 calculate final amount paid
+3 returns instead of printing

+2 correct string formatting with 2 decimal places

Question 5: Code Writing – getRange 30 points

Write a function that meets the following specifications:
Name: getRange
Parameters: a list of items, some of which are integers and floats (you may not assume that all list elements are numbers!)
Return: the difference between the largest and smallest numbers in the list as a floating point value

E.g: if the input list is [10,7,"Bob",23,21,3,8] the range would be 20.0 (23-3)

```
def getRange( aList):
  #Remove non-numbers from list.
  newList = []
  for item in aList:
    if type(item) = = int or type(item) = = float:
      newList.append(item)

  smallest  = min(newList)
  largest = max(newlist)

  range = largest - smallest

  return float( range)
```

Grading:
+10 - Removing or avoiding the non-numbers in the list.
+5 finding smallest (number)
+5 finding largest (number)
+5 calculating the difference between them
+5 returning a float!