Do not open this exam until you are told. Read these instructions:

1. This is a closed book exam. No calculators, notes, or other aids are allowed. If you have a question during the exam, please raise your hand.
2. You must turn in your exam immediately when time is called at the end.
3. 7 problems which add up to 100 points total. 1 hour 15 minutes. Each question’s point value is indicated.
4. In order to be eligible for as much partial credit as possible, show all of your work for each problem, write legibly, and clearly indicate your answers. Credit cannot be given for illegible answers.
5. After the last page there is paper for scratch work. If you need extra scratch paper after you have filled these areas up, please raise your hand. Scratch paper must be turned in with your exam, with your name and ID number written on it, but scratch paper will not be graded.

6. Fill in the following:

   NAME :

   G# :
1. For the following expressions, write what value and type they would evaluate to in the python interpreter, or state that an error/exception is raised. If an exception is raised, you don’t need to give a value – just give the type of exception. (35 points)

<table>
<thead>
<tr>
<th>Expression</th>
<th>Value</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. 7.0 + 3</td>
<td>_____</td>
<td>________________</td>
</tr>
<tr>
<td>b. 5 / 2</td>
<td>_____</td>
<td>________________</td>
</tr>
<tr>
<td>c. &quot;2&quot; + &quot;3&quot;</td>
<td>_____</td>
<td>________________</td>
</tr>
<tr>
<td>d. 1 in [1, 2]</td>
<td>_____</td>
<td>________________</td>
</tr>
<tr>
<td>e. '2 + 3'</td>
<td>_____</td>
<td>________________</td>
</tr>
<tr>
<td>f. range(3)</td>
<td>_____</td>
<td>________________</td>
</tr>
<tr>
<td>g. len([])</td>
<td>_____</td>
<td>________________</td>
</tr>
<tr>
<td>h. str(&quot;1&quot;)</td>
<td>_____</td>
<td>________________</td>
</tr>
<tr>
<td>i. 7 / 2.0</td>
<td>_____</td>
<td>________________</td>
</tr>
<tr>
<td>j. 5 % 2</td>
<td>_____</td>
<td>________________</td>
</tr>
<tr>
<td>k. int(&quot;23&quot;)</td>
<td>_____</td>
<td>________________</td>
</tr>
<tr>
<td>l. 2 = x</td>
<td>_____</td>
<td>________________</td>
</tr>
</tbody>
</table>
m. x = 3
x+=1
x

n. x + y

o. x = 2
x==x

p. 7 / 2 + 1

q. not False

r. 1 or False
2. Write python code below that prompts the user to enter a character with the prompt “Enter char:”. Then, decide whether the character’s integer value is greater than or less than 64. If it is less than 64, print out “SMALL” followed by the integer value of the character. If it’s greater than or equal to 100, print out “LARGE”, followed by the integer value of the character. Otherwise, print out “MEDIUM”. (10 points)

3. What is the output of the following code, assuming the user enters a list [13, 11, -1]? (8 points)

```python
list = input()
print list
for i in list:
    print i
    if i > 12:
        print "more than 12"
    if i > 10:
        print "more than 10"
    else:
        print "small"
```
4. What is the output of the following code? (17 points).

```python
found = False
no = False
list = ['a', 4, 4.0, 'cat', no, 6]
ctr = 0

while not found:
    print "elt: " + str(list[ctr])
    if list[ctr] == False:
        found = True
        list[ctr] = str(list[ctr]) + "1"
    if ctr < len(list) - 1:
        print list[ctr] == list[ctr+1]
    ctr = ctr + 1

print ctr
print list
```
5. What does the following code output (assume we run it in the python module mode as test.py, where all the code below is inside that one file): (15 points)

```python
print "exam 1"

def func1():
    print "func1"
    return 'hello'

def func2(x):
    print x
    return x

def func3(x):
    x = 4
    print "func3"
    print x

def main(x,y):
    print (x + y)
    return (x + y)

y = func1()
print y
print func1()
y = func2(2)
print y
y = func3(2)
print y
y = main(5,2)
print y
print main(2,3)
func2(5)
```
6. True/False – circle all that are true, and document your assumptions. (15 points)

   a. Variable names cannot start with a number in Python.

   b. If a function call does not have argument, you can leave off the parentheses.

   c. Python determines the type of a variable on-the-fly at runtime. That is, the type of a variable can change throughout the program.

   d. An empty list evaluates to False when used with a conditional such as an if statement.

   e. It is possible to call a built-in function such that you can call it with a variable number of parameters.

   f. The python can be run in an interpreter and module mode.

   g. I write my name on all exams
Scratch paper