Print the following very clearly

<table>
<thead>
<tr>
<th>Name:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Email id:</td>
</tr>
<tr>
<td>Pledge:</td>
</tr>
</tbody>
</table>

Notices
- Based on your past educational achievements, I expect you to do well on this test.
- Answer the questions in any order that you want.

Test rules
- Check before you leave the room, that you uploaded all of your solutions. Do not ask afterwards whether you can submit a forgotten solution.
- This pledged exam is closed notes. The only device you may access during the test is your laptop.
- Uploading after you leave the room means you withdrawing from the class with a test score of 0.
- Any cheating can result in failing the class and the incident being referred to the Honor Committee.
- Do not access class examples artifacts, web solutions, or your own past assignments during the test; that is, the only code you may access or view are ones that you develop for this test.
- The only windows allowed on your laptop are PyCharm and a single browser with tabs reachable from class website.

PyCharm
- PyCharm can be used for developing the programs to be submitted. It cannot be used for the short answer questions.

Short answers
- Whenever a written answer is a string, surround the string with single quotes. Also, make sure the uppercase and lowercase letters are easily distinguishable.
- Whenever a written answer is a list, surround the elements with a pair of brackets. The elements in the list should be separated by commas.
- Whenever a written answer is a decimal, include at least one digit after the decimal point.
- Whenever a written answer is an integer, do not include a decimal point.

Programs
- Programs should follow class programming practices; e.g., header comments, whitespace, identifier naming, etc.
- Whether a program is runnable is important.
- Only output what is requested.
Part 1: Short answers (20 points)

1. Suppose the following definitions are in effect

   \[
   \begin{align*}
   x &= 7 \\
   y &= 2 \\
   s &= 'quack' \\
   t &= 'duck power' \\
   u &= t.strip() \\
   v &= [2, 7]
   \end{align*}
   \]

   a. What is the value of \((x + x // y)\)?

   b. What is the value of \('s' + s\)?

   c. What is the value of \(s[2:4]\)?

   d. What is the length of \(u\)?

   e. What is the value of \(v\) after function invocation \(v.append('2')\)?

   f. What is the last value printed by the following code segment?

   ```python
   for r in range(1, 6):
       for c in range(1, 4):
           product = r * c
           print(product)
   ```
g. What happens when Python executes the following assignment statement? Briefly explain.
\[ f1, f2 = \text{float('15.3 25.6')} \]

h. Consider the following code segment.
```python
import random
random.seed(0)
n1 = random.randrange(1, 5)
random.seed(0)
n2 = random.randrange(1, 5)
print(n1, n2)
```
Can the output print two different values? Why.

i. Which of the following statements are not Python style rules?
1) Multiplication has higher precedence than addition.
2) Identifiers begin with an alphabetic character
3) Identifiers are written in snake_case.

j. Consider the following gotcha code segment.
```python
for n in (1, 5):
    print(n)
```
Does the segment print four values? Why.
Part 2: Programming (80 points)

2. Implement program `buses.py`. The program separately prompts and reads respectively two integer values – the number of people \( p \) and the number of buses \( b \). The program computes and prints the decimal average number of people per bus. No other output is to be given.

Two sample program runs are given below

Enter number of people: 315
Enter number of buses: 10
31.5

Enter number of people: 144
Enter number of buses: 5
28.8

3. Implement program `soup.py`. The program has a single prompt for respectively two decimal values – the radius \( r \) and height \( h \) of a soup can. The program computes and prints the volume of the soup can; that is, \( \pi r^2 h \).

Two sample program runs are given below.

Enter radius and height: 1.25 3.25
15.953400194010669

Enter radius and height: 1.5 6.75
47.712938426394985

4. Implement program `lwa.py`. The program prompts for a single line of text. The program prints one line of output with three values. They are respectively the amount of text entered, the number of words in the text, and the number of a’s in the text.

Two sample program runs are given below.

Enter text: here is to the hard-working people
34 6 1

Enter text: here is to the salt of the earth
32 8 2

5. Implement program `glue.py`. The program prompts for a line of text. The program prints the words in the text concatenated (added together) with each word followed by a single hyphen.

Two sample program runs are given below.

Enter text: i'm the ocean
i'm-the-ocean-

Enter text: on the long plain
on-the-long-plain-
6. Implement program `reveal.py`. The program separately prompts for the name of a CS 1112 web file and then for a list of indices. The web file can be found in the folder:

   http://www.cs.virginia.edu/~cs1112/text/

The program prints on separate lines, those words in the web file indicated by the list of indices.

Two sample program runs are given below.

```
Enter data file: macaronic.txt
Enter indices: 8 3 2 6 4 1 0 9 5 7
macaronic
denotes
a
mixture
of
words
drawn
from
different
languages
```

```
Enter data file: scramble.txt
Enter indices: 13 15 34 1 4 24 9 27 35 4 31 12 8 14
want
what
you
do
.
do
what
you
want
.
but
always
yabba-dabba-doo
do
```

For your information, the current contents of `macaronic.txt` is
```
drawn words a denotes of different mixture languages macaronic from
```

For your information, the current contents of `scramble.txt` is
```
gone do good number . since one been yabba-dabba-doo
what is can always want do what no is be will
just one it that do as you you have ever two but
the bad you want loneliest you
```