1. (40 points)

Consider the following Python program that computes the sum of first $n$ odd numbers and verifies that the sum is equal to $n^2$.

```python
# Implements
# Sum of first n odd numbers is n^2
#
import operator

n = input('Pick a number greater than or equal to 0 for an upper limit: ')  
print ('You entered n = %s. \n' % n)
	pair = [1, 1]  # the basis pair
print ('pair is %s \n' % pair)

for i in range (1,n+1):  # executes for i = 1, 2, ..., n
    pair[0] = pair[0] + 1
    pair[1] = pair[1] + pair[0]*2 - 1
    print ('pair is %s \n' % pair)

if operator.eq (pair[1],pair[0]**2):
    print "True: the second member of the pair is the square of the first."
else:
    print "False: the second member of the pair is not the square of the first."
```

Prove using induction that the program always prints

"True: the second member of the pair is the square of the first."

You must present the proof based on the code and on the number of iterations of the for loop. Clearly label the basis, inductive hypothesis, and inductive step.

Please flip the page for additional questions.
2. (30 points)
Consider the set $L_1$ of strings over $\Sigma = \{1, 2, a, b, c\}$ that start with “a” and end with “1”.
Strings can have a length of two or greater.
(a) Give a recursive definition for $L_1$.
(b) Give a regular set for $L_1$.

3. (30 points)
Consider the set $L_2$ of strings over $\Sigma = \{1, 2, a, b, c\}$ that start and end with a number.
Strings can have a length of one or greater.
(a) Give a recursive definition for $L_2$.
(b) Give a regular set for $L_2$. 