Evan Wilcox
CS1200
Homework 1
Due: 2018-09-04, 11:59 PM

1. Simple questions about python.
    a.

```
C:\Users\Evan\Documents\MST Student Drive\CS1200>python hw01.py
you hop!
you must trip yourself
```

    b.

```
C:\Users\Evan\Documents\MST Student Drive\CS1200>python hw01.py
a
o
fo
fo
foa
raboof
```

    c.

```
C:\Users\Evan\Documents\MST Student Drive\CS1200>python hw01.py
b bbbb c cccc
```

    d.

```
C:\Users\Evan\Documents\MST Student Drive\CS1200>python hw01.py
[[1, 2, 3], [1, 2, 3]]
```

    e.

```
46  #1e
47  for i in range(2, 11):
48      print "1/%s = %.3f" % (i, 1.0/i)
```

```
C:\Users\Evan\Documents\MST Student Drive\CS1200>python hw01.py
1/2 = 0.500
1/3 = 0.333
1/4 = 0.250
1/5 = 0.200
1/6 = 0.167
1/7 = 0.143
1/8 = 0.125
1/9 = 0.111
1/10 = 0.100
```

2. Fermat's Last Theorem

a.

```
51  #2a
52  def fermatCheck(a, b, c, n):
53      if ((a**n + b**n) == c**n) and n > 2:
54          print "Fermat was wrong!"
55      else:
56          print "That doesn't work"
57
58  fermatCheck(3987, 4365, 4472, 12)
```

b.

```
C:\Users\Evan\Documents\MST Student Drive\CS1200>python hw01.py
That doesn't work
```

3. altDif()

```
61  #3
62  def altDif(nums):
63      if len(nums) <= 1:
64          return nums[0]
65      else:
66          return nums[0] - altDif(nums[1:])
67
68
69  myNums1 = [7, 6, 9, 10]
70  myNums2 = [3, 5, 4]
71  myNums3 = [6, 7]
72
73  print altDif(myNums1)
74  print altDif(myNums2)
75  print altDif(myNums3)
```

```
C:\Users\Evan\Documents\MST Student Drive\CS1200>python hw01.py
0
2
-1
```

4. f()
   a.

```
C:\Users\Evan\Documents\MST Student Drive\CS1200>python hw01.py
92
92
92
92
92
241
```

   b.

```
 95  #4b
 96  def g(x):
 97      if x <= 100:
 98          return 92
 99      else:
100          return x - 9
```

5. SuperReverse()

```
110  #5
111  def SuperReverse(l):
112      if len(l) <= 1:
113          return l
114      else:
115          return SuperReverse(l[-1:]) + SuperReverse(l[:-1])
116
117  mylist = [[1,9], [5, [6, 2]], 3]
118  print SuperReverse(mylist)
```

```
C:\Users\Evan\Documents\MST Student Drive\CS1200>python hw01.py
[[1, 9], [5, [6, 2]], 3]
[3, [[2, 6], 5], [9, 1]]
```

6. The Ackermann Function

a.

```
C:\Users\Evan\Documents\MST Student Drive\CS1200>python hw01.py
Q(0,0) = 1
Q(0,1) = 2
Q(0,2) = 3
Q(0,3) = 4
Q(0,4) = 5
Q(0,5) = 6
Q(1,0) = 2
Q(1,1) = 3
Q(1,2) = 4
Q(1,3) = 5
Q(1,4) = 6
Q(1,5) = 7
Q(2,0) = 3
Q(2,1) = 5
Q(2,2) = 7
Q(2,3) = 9
Q(2,4) = 11
Q(2,5) = 13
Q(3,0) = 5
Q(3,1) = 13
Q(3,2) = 29
Q(3,3) = 61
Q(3,4) = 125
Q(3,5) = 253
```

Maximum recursion depth reached when X = 4.

b. For Q(2,2), Q is called 27.