CIS 125 Principles of Programming Logic
Repetition (Looping) Structures

Further and Advanced Information

For Loops

Example: Nested for loop

Nested loops are common. They are a loop within a loop. Here is one example.

```
while True:
    num = int(input("Enter a max value: (< 1 value to exit): "))
    if num < 1:
        break
    for x in range(num):
        print(x+1)
```

Output:

```
Enter a max value: (< 1 value to exit): 4
1
2
3
4
Enter a max value: (< 1 value to exit): -1
```

Example: Nested for loop #2

```
for x in range(1, 4):
  for y in range(1, 3):
          print (x, '*', y, '=', x*y )
```

Output:

1 * 1 = 1
1 * 2 = 2
2 * 1 = 2
2 * 2 = 4
3 * 1 = 3
3 * 2 = 6

Example: Loop that calls functions with decision statement and recursion

Below is an example of recursion. Recursion is when a function calls itself from within itself. In this program, if the user types in an invalid test score (i.e. on below 0 or above 100), the function is called again, i.e. score = **getScore()**

```
def getScore():
    score = int(input("Please enter test score #" + str(x) + ": "))
    if score < 0 or score > 100:
        print("Invalid number. Please re-enter.")
        score = getScore()
    return score

# Main
total = 0
max = int(input("Enter number of exam scores to average: "))
for x in range(1,max+1):
    score = getScore()
    total += score
print("Class average = %.2f " % (total/max))
```

Sample run (user inputs – 99 88 77):

Enter number of exam scores to average: 3
Please enter test score #1: 333
Invalid number. Please re-enter.
Please enter test score #1: 99
Please enter test score #2: 88
Please enter test score #3: 77
Class average = 88.00

Example: Looping through list

```
fruits = ['oranges', 'pears', 'grapes', 'apples']
for fruit in fruits:
   print ("I like %s" % fruit)
```

Output:

I like oranges
I like pears
I like grapes
I like apples

Example: for loop that accumulates from a list with different data type output

```
list = [5,10,20,30]
sum = 0
for num in list:
   sum = sum + num
print ("The sum is: %d" % sum)
print ("The sum is: %f" % sum)
print ("The sum is: ",  sum)
```

Output:

```
The sum is:   65
The sum is:    65.000000
The sum is:      65
```

Example: Loop through a string

```
string = "Hello"
for x in string:
        print (x)
```

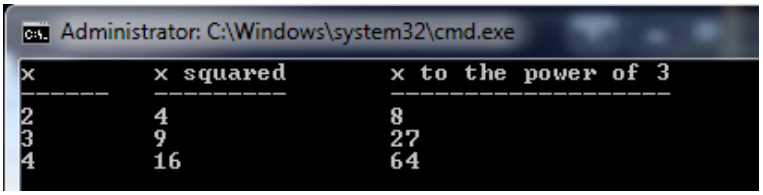
Output:

```
H
e
l
l
o
```

Example: For loop with tabs and Windows OS clear screen

```
import os
os.system('cls')
print ("x \t x squared \t x to the power of 3")
print ("------ \t --------- \t ------------------- \t")
for x in range(2, 5):
        print (x, '\t', x**2, '\t\t', x**3 )
```

Output:



Example: For loop with else related to for using a list and nested if statement

```
at_home = ["oranges", "olives","bananas","pickles"]
for food in at_home:
        if food == "anchovies":
                print("No more anchovies please!")
                break
        print("Great. I really like " + food)
else:
        print("Fantastic - no anchovies!")
print("I'm full now.")
```

Output:

```
Great. I really like oranges
Great. I really like olives
Great. I really like bananas
Great. I really like pickles
Fantastic - no anchovies!
I'm full now.
```

Example: For loop with decrementer (bottles of water on a wall)

```
for i in range(99, 0, -1):
  if i == 1:
     print('1 bottle of water on the wall.')
     print('Take it down, pass it around, no more bottles of water.')
  else:
     print(i, 'bottles of water on the wall')
     print('Take one down, pass it around,', (i - 1), 'more bottles of water.')
```

Example: For loop inside function with accumulator (total) and returning two variables

```
def loopFunction(total):          # function returns two values
        num = int(input("How many times do you want to loop? "))
        for x in range(num):
                total = total + x
                print("x=", x)
        return num, total

total = 0
print("Before loop: x=", x, "and total =", total)
x, total = loopFunction(total)
print("After loop: x=", x, "and total =", total)
```

Output:

```
Before loop: x= 3 and total = 0
How many times do you want to loop?  4                    {user input}
x= 0
x= 1
x= 2
x= 3
After loop: x= 4 and total = 6
```

While Loops

Example: While loop incrementing by more than 1

```
counter = 1
max = 20
while counter <= max:
        print ("Counter =", counter)
        counter += 4
```

Output:

```
Counter = 1
Counter = 5
Counter = 9
Counter = 13
Counter = 17
```

Example: while loop with user input nested if statement

```
i = 1
x = int(input("How high do you want to count to? "))
while (i <= x):
    print (i)
    i += 1
    if i > 10:
        print("Sorry, we only allow you to count to 10. The End.")
        break
```

Output:

```
How high do you want to count to:  4
1
2
3
4
```

Example: Loop in a user-defined function

```
def counterLoop(x):
   while x >= 0:
      print(x)
      x-=1

x = 3
counterLoop(x)
```

Output:

```
3
2
1
0
```

Example: Loop that calls function based on user input

```
def counterLoop(x):
   total = x * 10
   print("Total: ", total)

# Main Program
i = 1
x = int(input("Please enter a number: "))
while i <= x:
   counterLoop(i)
   i+=1
```

Output:

```
Total: 10
Total: 20
Total: 30
Total: 40
```

Example: Four ways to open external files in Python

```
import webbrowser
webbrowser.open_new(r'periapical-lesion-diagnosis-flowchart.pdf')

import subprocess
subprocess.Popen("periapical-lesion-diagnosis-flowchart.pdf",shell=True)

import os
os.startfile("periapical-lesion-diagnosis-flowchart.pdf")

# Mac syntax:  os.system('open <myFile>')
```

Example: Running a function in another Python file (user-defined library)

Filename: program1.py

```
import my_library
num = int(input("Please enter max number: "))
my_library.displayText(num)
```

Filename: my_library.py

```
def displayText(num):
      for x in range(num):
         print("Count: ", x)
```

Sample run:

```
Please enter max number: 3
Count: 0
Count: 1
Count: 2
```