1.  
This program has one variable and a list  
count = -11  
a\_list=[]  
count is used in determine while loop continuation or termination  
append count to a\_list  
on loop exit, display a\_list

Output:  
[-11, -10, -9, -8, -7, -6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

2.  
Th program has two lists and a variable, num  
ListA = [1,2,3,4,5]  
ListB = []

In a for loop with num and listA  
append num to listB  
on loop exit, assign 10 to the third element of listB  
display the third element of listA and listB  
display listB

Output:  
10  
3  
[1, 2, 3, 4, 5]

3.  
This program has a variable, num and i, and to lists  
ListA = [1,2,3,4,5]  
ListB = []

In a for loop with num and listA  
decrement i by 8  
append i to listB  
on loop exit, assign 10 to the third element of listA  
display the third element of listA and listB  
display listB

Output:  
10  
-5  
[-7, -6, -5, -4, -3]

4.   
This program has one list and two variables  
L3=[ ]  
with range() argument 5 in a for loop  
square the value returned from range() and append() it to L3  
Display L3 on loop exit

Output:  
[0, 1, 4, 9, 16]

5.  
This program has one list and variable  
L4=[ ]  
count = 0  
loop in a while loop to less than 5  
append() count to L4  
Display L4 on loop exit

Output:  
[0, 1, 2, 3, 4, 0, 1, 2, 3, 4]  
  
6.  
This program has one list  
list1=[4,8,9,6,5,4,8,7,8]  
compute and display is average

7.  
This program has two lists  
ListA = [1,2,3,4,5]  
ListB = []

A for loop uses num for control, along with ListA  
in the for loop, append num to ListB   
outside the loop, assign 10 to the 3rd element of ListA; print this element

Print ListB

Output:  
3

[1, 2, 3, 4, 5]

8.  
In a for loop with range() this program displays 7 11 13; range() starts at 15 and the values returned in the for is squared along with other arithmetic operations

9.  
This function takes one string  
in a for loop, find the last k in the string  
displays the index after k is found

String: Kim KPark k

10.  
For the first input() the user enters 6  
In the while loop the values entered must not be in the range -1 and +1

What is the output if each of the other values entered is the number 2   
Demonstrate valid and invalid ranges

istart = 1

#jstop = 5

kstep = 1

sum = 0

jstop = int(input('enter th number of values to read =>'))

while istart <= jstop:

range(istart, jstop, kstep)

print('enter')

get\_int = int(input())

istart = istart + kstep

??????????

print('The value enterd is not valid')

else:

sum = sum + get\_int

print('on exit', istart)

print('sum ', sum)

Output:

on exit 7

sum 12

11.  
This program has   
one list with these values: 2, 4, 3, 5, 6, 5, 7, 8, 9, 11, 3  
two while loops  
Two variables j and k  
J controls the execution of the outer loop, k the inner  
a Boolean variable dups  
  
You are determine if there are any duplicates in the list

code\_list=[2, 4, 3, 5, 6, 5, 7, 8, 9, 11, 3]

dups = False

j = 0

while j < len(code\_list)? and not dups:

?('j = ', j)

k = j + 1

while k < len(code\_list) and not dups:

if code\_list[k] ? code\_list[j]:

dups = True

print('k = ', k)

k ?

j ?

print('dups = ', dups)  
  
Output:  
j = 0

j = 1

j = 2

k = 10

dups = True

12.

This program has   
one function  
a for loop with i as its control  
a variable j = 0

In the for loop, j is assigned to the jth index of list  
this list passed to refyn() has values 3, 5, 6, 2  
display list before the call to refyn()

display list after the loop exits in refyn()

display list after the call to refyn()

**Explain** the results

def refyn():

j = 0

for i in refyn\_list\_vals:

refyn\_list\_vals[j] = j

j = j + 1

print('refyn\_list\_vals before exiting refyn() = ', refyn\_list\_vals)

print('list\_vals before call to refyn = ', list\_vals)

refyn(list\_vals)

print('list\_vals after call to refyn = ', list\_vals)

Output:  
list\_vals before call to refyn = [3, 5, 6, 2]

refyn\_list\_vals before exiting refyn() = [0, 1, 2, 3]

list\_vals after call to refyn = [0, 1, 2, 3]

13.  
This program has   
tup = ((2, 3, 4, 5), (6, 7, 8))

while loops

and displays:  
2

3

4

5

next tuple follows:

6

7

8

14.  
This program has   
a 4-element tuple that needs and additional element to make it a 5-element tuple

f1()   
the takes one argument  
converts the argument to a list  
assign 20 to the 5th element of the list  
display the list  
return the list when finished

outside f1()  
tup = (2, 3, 4, 5)  
display the list  
display the tuple

Output:  
[2, 3, 4, 5]

[2, 3, 4, 5, 20]

(2, 3, 4, 5, 20)

15.  
This program has one dictionary named cities  
display cities  
if FL in cities, delete it, del cities['FL']  
otherwise, include FL and city Tallahassee  
display cities

Output:  
{'GA': 'Atlanta', 'NY': 'Albany', 'CA': 'San Diego'}

{'GA': 'Atlanta', 'NY': 'Albany', 'CA': 'San Diego', 'FL': 'Tallahassee'}

16.  
This program prompts the user to enter a directory and file name to create  
Write your name to the newly created file in the newly created directory

17.  
Call f1() 3 times to produce the outputs below; what are the calls that produce them?

def f1(lis):

for i in range(2, len(lis)):

lis[i] = lis[i] + lis[i - 1] + lis[i - 2]

Outputs:

4 8 19

3 7 14 26

1 13 16

18.  
Call f1() 4 times to produce the outputs below; what are the calls that produce them?

def f1(list):

for i in range(0, len(list) - 1):

if i % 2 == 0:

list[i] += 1

else:

list[i] -= 1

print(list[i])

Outputs:

3

5   
7  
22

Thanks  
Winston