Security and Networking with Python (CSE 2157)

MINOR ASSIGNMENT-2: Functions and Control Structures

1. What will be the output produced by each of the following function calls:
   i. math.ceil(65.65)
   ii. math.ceil(65.47)
   iii. math.fabs(-67.58)
   iv. math.fabs(3)
   v. math.exp(2.7)
   vi. math.log(45,2)
   vii. math.log10(1000)
   viii. math.pow(4,1/2)
   ix. math.sqrt(121)
   x. math.radians(30)
   xi. math.degrees(math.pi/2)

2. Give the range in which value of variable x may lie onexecution of the following statements:

   ```
   import random
   x = random.random() + 5
   ```

3. Consider the following function:

   ```
   def nMultiple(a = 0, num = 1):
   return a * num
   ```

   What will be the output produced when the following calls are made:

   i. nMultiple(5)

   ii. nMultiple(5, 6)

   iii. nMultiple(num = 7)

   iv. nMultiple(num = 6, a = 5)

   v. nMultiple(5, num = 6)

4. Study the program segments given below. Give the output produced, if any.

i. def test(a, b):

```
        a = a+b
        b = a-b
        a = a-b
        print('a = ', a)
        print('b = ', b)
test(5,8)
```

ii.def func():

```
 pass
a = func()
print(a)
```

5. Write a function *areaTriangle* that takes the lengths of three sides: side1, side2, and side3 of the triangle as the input parameters and returns the area of the triangle as the output. Also, assert that sum of the length of any two sides is greater than the third side. Write a function main that accepts inputs from the user interactively and computes the area of the triangle using the function *areaTriangle*.

6. Write an assignment statement using a single conditional expression for the following if-else code:

```
if marks >=70:
 remarks = 'good'
else:
 remarks = 'Average'
```

7. Study the program segments given below. In each case, give the output produced, if any.

i.
```
total = 0
count = 20
while count > 5:
 total += count
 count -= 1
print(total)
```

ii.
```
total = 0
N = 5
for i in range (1, N+1):
  for j in range (1, i+1):
      total += 1
print(total)
```

iii.
```
total = 0
N = 10
for i in range (1, N+1):
  for j in range (1, N+1):
```

```
            total += 1
        print(total)
```

iv.
```
total = 0
N=5
for i in range (1, N+1):
  for j in range (1, i+1):
   total += 1
  total -=1
 print(total)
```

v.
```
total = 0
N=5
for i in range(1, N+1):
 for j in range(1, N+1):
  total += i
 print(total)
```

vi.
```
total = 0
N=5
for i in range(1, N+1):
   for j in range(1, i+1):
      total += j
 print(total)
```

vii.
```
total = 0
N=5
for i in range(1, N+1):
  for j in range(1, N+1):
      total += i + j
print(total)
```

viii.
```
total = 0
N=5
for i in range(1, N+1):
  for j in range(1, i+1):
   for k in range(1, j+1):
      total += 1
 print(total)
```

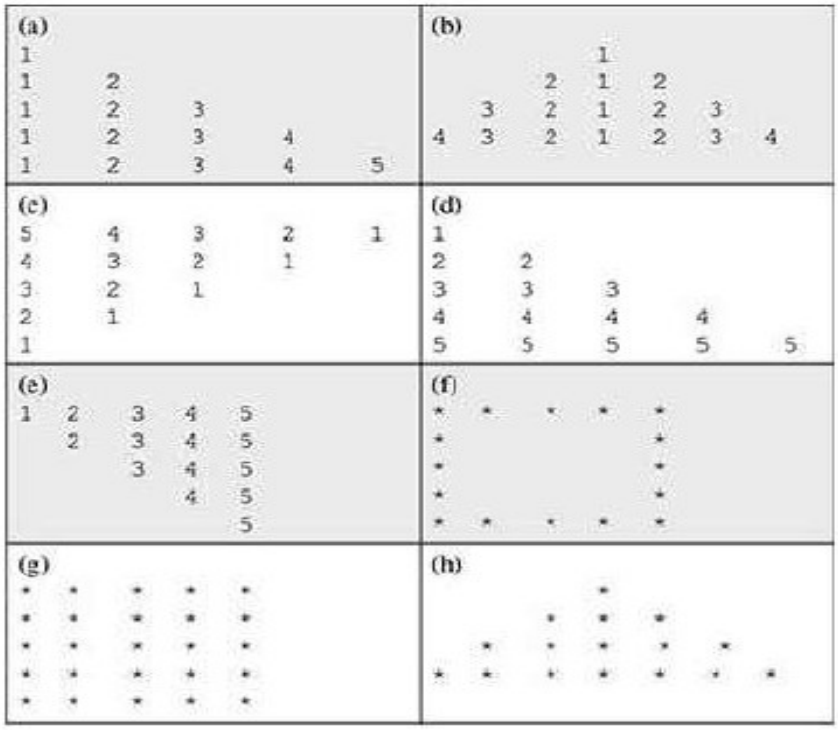
ix.
```
number = 72958476
a, b = 0, 0while (number > 0):
digit = number % 10
   if (digit % 2!= 0):
      a += digit
   else:
      b += digit
   number /= 10
 print(a, b)
```

8. Write a function to determine whether a given natural number is a perfect number. A natural number is said to be a perfect number if it is the sum of its divisors. For example, 6 is a perfect

number because 6=1+2+3, but 15 is not a perfect number because 15≠1+3+5.

9. Write a function that takes two numbers as input parameters and returns their greatest common divisor.

10. Write a function that accepts as an input parameter the number of rows to be printed and prints a figure like:



11. Write a function that finds the sum of the n terms of the following series:

i. 1 – x2 /2! + x4 /4! – x6 /6! + … xn /n!

ii. $e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$

12. Write a function that returns True or False depending on whether the given number is a palindrome.

13. Write a function that returns the sum of digits of a number, passed to it as an argument.

14. Write a program that prints Armstrong numbers in the range 1 to 1000. An Armstrong number is a number whose sum of the cubes of the digits is equal to the number itself. For example, 370 = 3 + 7 + 0.

15. Write a function that takes two numbers as input parameters and returns True or False depending on whether they are co-primes. Two numbers are said to be co-prime if they do not have any common divisor other than on.