Programming in Python (CSE 3142)

MINOR ASSIGNMENT-1: BASIC ELEMENTS OF PYTHON PROGRAMMING

1. Evaluate the following expressions:
   (x<y) or (not(z==y) and (z<x))
   a. x =0, y=6, z=10
   b. x=1, y=1, z=1

2. Evaluate the following expressions involving arithmetic operators:
   a. -7*20+8/16*2+54
   b. 7**2//9%3
   c. (7-4*2)*10-25*8//5
   d. 5%10+10-25*8//5
   e. ’hello’*2-5

∧

3. Evaluate the following expressions involving relational and logical operators:
   a. ’hi’ > ’hello’ and ’bye’ < ’Bye’
   b. ’hi’ > ’hello’ or ’bye’ < ’Bye’
   c. 7 > 8 or 5 < 6 and ’I am fine > ’I am not fine’
   d. 10 !=9 and 29 >= 29
   e. 10 !=9 and 29 >= 29 and ’hi’ > ’hello’ or ’bye’ < ’Bye’ and 7 <= 2.5

4. Evaluate the following expressions involving arithmetic, relational and logical operators:
   a. 5 % 10 + 10 < 50 and 29 >= 29
   b. 7 ** 2 <= 5 // 9 % 3 or ’bye’ < ’Bye’
   c. 5 % 10 < 8 and -25 > 1 * 8 // 5
   d. 7 ** 2 // 4 + 5 > 8 or 5 != 6
   e. 7/4 < 6 and ’I am fine > ’I am not fine’
   f. 10 + 6 * 2 ** 2 != 9//4-3 and 29 >= 29/9
   g. ’hello’ * 5 > ’hello’ or ’bye’ < ’Bye’

5. Evaluate the following expressions involving bitwise operators: a. 15 & 22
   b. 15 | 22
   c. -15 & 22
   d. -15 | 22
   e. ~ 15
   f. ~ 22
   g. ~ -20
   h. $15 \ \ 22$
   i. $8 << 3$
   j. $40 >> 3$

6. Differentiate between the following operators with the help of examples:
   a. = and ==
   b. / and %
   c. / and //
   d. * and **

7. What output will be displayed when the following commands are executed in Python shell in se- quence:

a.
```
>>> a = 6
>>> a == 6
>>> a < 5.9
>>> a > 5.9
```

b.
```
>>> b = 7
>>> b / 6
>>> b // 6
>>> b / 4
>>> b % 4
>>> b % 7
>>> b * 2
>>> b ** 2
```

8. Construct logical expressions for representing the following conditions:
    a. marks scored should be greater than 300 and less than 400.
    b. Whether the value of grade is an uppercase letter.
    c. The post is engineer and experience is more than four years.

9. Write Python statements for the following equations:

a. root1= $\dfrac{-b+\sqrt{b^2-4ac}}{2a}$

b. result= $\dfrac{2xy-9y}{2xy^3}-\dfrac{4yx^2}{2y}$

c. result= $2\cos\frac{1}{2}(x+y)\cos\frac{1}{2}(x-y)+e^x-1-\frac{x}{4}+\tan x-\log(v)$

10. How does the effect of the following two statements differ?
    a. x += x + 10
    b. x = x+ 10

11. Write a program that asks the user to enter the width and length of a room. Once these values have been read, your program should compute and display the area of the room. The length and the width will be entered as floating-point numbers. Include units in your prompt and output message; either feet or meters, depending on which unit you are more comfortable working with

12. An online retailer sells two products : widgets and gizmos . Each widget weighs 75 grams . Each gizmo weighs 112 grams . Write a program that reads the number of widgets and the number of gizmos from the user . Then your program should compute and display the total weight of the parts

13. Write a program that determines how quickly an object is travelling when it hits the ground. The user will enter the height from which the object is dropped in meters (m). Because the object is dropped its initial speed is 0 m/sAssume that the acceleration due to gravity is $9.8m/s^2$. You can use the formula $v_f = \sqrt{(v_i^2 + 2ad)}$ to compute the final speed, v_f , when the initial speed, v_i , acceleration, a, and distance, d, are known.

14. Write a program that reads a four-digit integer from the user and displays the sum of its digits. For example, if the user enters 3141 then your program should display $3 + 1 + 4 + 1 = 9$.

15. Write a program that reads three integers from the user and displays them in sorted order (from smallest to largest). Use the min and max functions to find the smallest and largest values. The middle value can be found by computing the sum of all three values, and then subtracting the minimum value and the maximum value.

16. Create a program that reads duration from the user as a number of days, hours, minutes, and seconds. Compute and display the total number of seconds represented by this duration.

Name: Sunil Kumar Pradhan
Regn Number: 21401016300

Programming in Python (CSE 3142)
MINOR ASSIGNMENT-2: FUNCTIONS

1. What will be the output produced by each of the following function calls:
   a. math.ceil(65.65)
   b. math.ceil(65.47)
   c. math.fabs(-67.58)
   d. math.fabs(3)
   e. math.exp(2.7)
   f. math.log(45,2)
   g. math.log10(1000)
   h. math.pow(4, 1/2)
   i. math.sqrt(121)
   j. math.radians(30)
   k. math.degrees(math.pi/2)

2. Give the range in which value of variable x may lie on execution of the following statements:
   ```
   import random
   x = random.random() + 5
   ```

3. Evaluate the following expressions using Python shell. Assume that ASCII coding scheme is used for character data.
   a. abs(-5.4)
   b. abs(15)
   c. chr(72)
   d. round(-24.9)
   e. float(57)
   f. complex(’1+2j’)
   g. divmod(5,2)
   h. float(57)
   i. pow(9,2)
   j. max(97, 88, 60)
   k. min(55, 29, 99)
   l. max(’a’, ’b’, ’AB’)

4. 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:
   a. nMultiple(5)
   b. nMultiple(5,6)
   c. nMultiple(num = 7)
   d. nMultiple(num = 6, a = 5)
   e. nMultiple(5, num = 6)

5. Develop Python functions to produce the following outputs:

a.

```
        *
   *    *    *
*  *    *    *    *
   *    *    *
        *
```

b.

```
$ $ $ $ $
$       $
$       $
$       $
$ $ $ $ $
```

,

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

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

b.
```
def func():
        pass
    a = func()
    print(a)
```

7. 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.

8. Write a function to print the table of entered number.

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

a.
```
def say (message, times=2):
        print(message*times)
say (‘Hello’)
say(‘World’,5)
```

b.
```
def fun(a=2,b=3,c=7):
        d= a+b+c
        print(d)
print(fun (2))
```

10. Find the sum of even digits of a four-digit number using function.
Warning: Don’t use control structures.

11. Using a function evaluate the value of the arithmetic expression taken from the user.
Hint: Expression will act as an argument while defining function.

12. What does a function return by default in Python? Define a function that does not return any value, store the function call in a variable and check the value of that variable.

13. Write a function which takes co-ordinates of three points as input and returns true if points are collinear otherwise returns false.

14. Write a function named as ‘UpperCase’ which converts the lower case alphabet to uppercase alphabet. Also, assert that the entered alphabet by user is valid lowercase alphabet. Write a function main that accepts inputs from the user interactively and converts the lowercase alphabet to uppercase using the function ‘UpperCase’.

15. Observe carefully the below function

```
def fun(a=0, b=1):
    return (a**2 + b**2)
```

What will be the output for each call made below?

a.) fun(2,a=3)
b.) fun(b=3,2)
c.) fun(3,b=2)
d.) fun(a=4,5)

16. What will be the output of following code?

```
x = -5
def display(x):
    print(x)
    x = 5
    print(x)
display(x)
print(x)
```

a. -5, 5, -5
b. -5, 5, 5
c. -5, -5, -5
d. None of these.

17. What will be the output of the following Python code?

```
int('89.67')
```

a. Import Error
b. Value Error
c. Type Error
d. Name Error

18. Create the following scripts importedModule and mainModule in the working directory, execute the script mainModule and justify the output.

- importedModule.py

```
def test1():
    print(’test1 in imported module’)

def test2():
    print(’test2 in imported module’)

test1()
test2()
```

- mainModule.py

  ```
  import importModule
  print(’hello’)
  ```

Programming in Python (CSE 3142)
MINOR ASSIGNMENT-3: CONTROL STRUCTURES

1. Write an assignment statement using a single conditional expression for the following if-else code:
```
if marks >= 70:
        remarks = 'good'
else:
        remarks = 'average'
```

2. It is commonly said that one human year is equivalent to 7 dog years. However, this simple conversion must recognize that dogs reach adulthood in approximately two years. As a result, some people believe that it is better to count each of the first two human years as 10.5 dog years and then count each additional human year as 4 dog years. Write a program that implements the conversion from human to dog years described in the previous paragraph. Ensure that your program works correctly for conversions of less than two human years and conversions of two or more human years. Your program should display an appropriate error message if the user enters a negative number.

3. The length of a month varies from 28 to 31 days. Create a program that reads the month's name from the user as a string. Then your program should display the number of days in that month. Display "28 or 29 days" for February so that leap years is addressed.

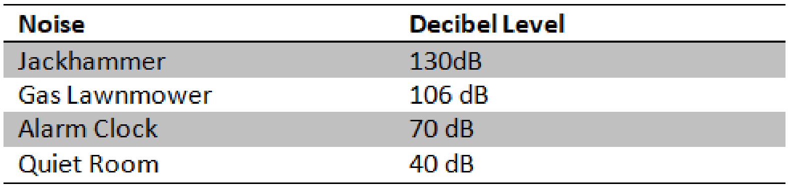
4. A roulette wheel has 38 spaces on it. Of these spaces, 18 are black, 18 are red, and two are green. The green spaces are numbered 0 and 00. The red spaces are numbered 1, 3, 5, 7, 9, 12, 14, 16, 18, 19, 21, 23, 25, 27, 30 32, 34 and 36. The remaining integers between 1 and 36 are used to number the black spaces. Many different bets can be placed in roulette. We will only consider the following subset of them in this exercise:

   - Single number (1 to 36, 0, or 00)
   - Red versus Black
   - Odd versus Even (Note that 0 and 00 do not pay out for even)
   - 1 to 18 versus 19 to 36

   Write a program that simulates a spin of a roulette wheel by using Python's random number generator. Display the number that was selected and all of the bets that must be payed. For example, if 13 is selected then your program should display:
```
The spin resulted in 13...
Pay 13
Pay Black
Pay Odd
Pay 1 to 18
```
   If the simulation results in 0 or 00 then your program should display Pay 0 or Pay 00 without any further output

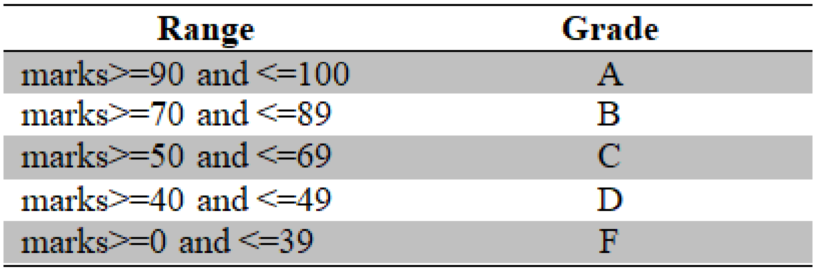
5. The following table lists the sound level in decibels for several common noises. Write a program that reads a sound level in decibels from the user. If the user enters a decibel level that matches one of the noises in the table then your program should display a message containing only that noise. If the user enters a number of decibels between the noises listed then your program should display a message indicating which noises the value is between. Ensure that your program also generates reasonable

output for a value smaller than the quietest noise in the table, and for a value larger than the loudest noise in the table.



Noise	Decibel Level
Jackhammer	130dB
Gas Lawnmower	106 dB
Alarm Clock	70 dB
Quiet Room	40 dB

6. Write a python script to assign a grade to a student based on marks obtained as per the criteria mentioned in the above table:



Range	Grade
marks>=90 and <=100	A
marks>=70 and <=89	B
marks>=50 and <=69	C
marks>=40 and <=49	D
marks>=0 and <=39	F

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

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

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

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

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

e.
```
total = 0
N = 5
for i in range(1, N+1):
  for j in range(1, N+1):
       total += i
   print(total)
f. total = 0
   N = 5
   for i in range(1, N+1):
     for j in range(1, i+1):
       total += j
   print(total)
g. total = 0
   N = 5
   for i in range(1, N+1):
     for j in range(1, N+1):
       total += i+j
   print(total)
h. 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)
i. number = 72958476
   a, b = 0, 0
   while (number > 0):
   digit = number % 10
     if(digit % 2 != 0):
       a += digit
     else:
       b += digit
     number /= 10
   print(a,b)
```

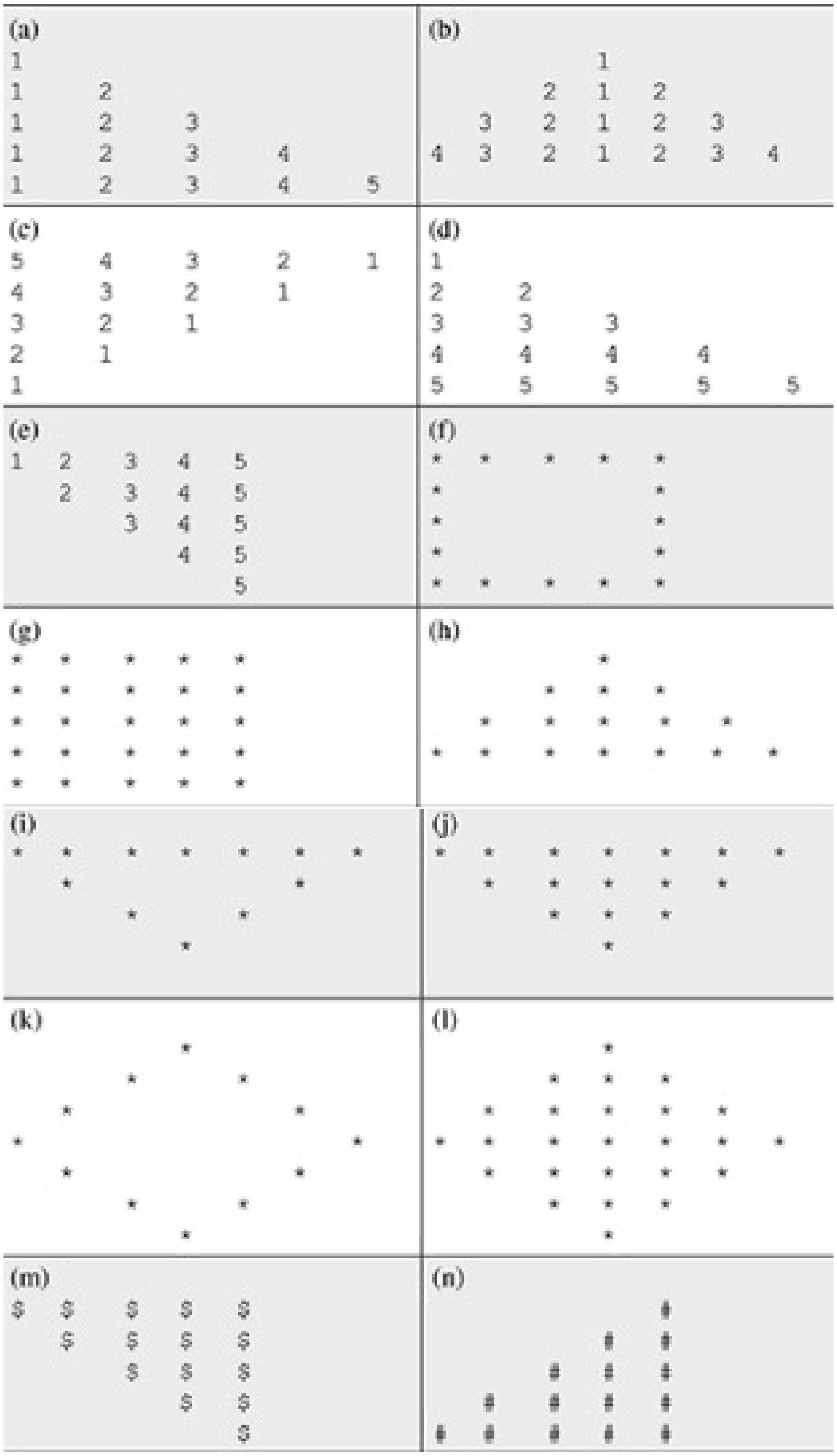
8. Write a function to determine whether a given natural number is perfect. A natural number is said to be a perfect number if it is the sum of its divisors. For Example, 6 is a perfect number because 6 = 1+2+3, but 15 is not because 15 != 1+3+5.

9. Write a program to find the maximum of three numbers using a nested function.

10. Write a function that takes two numbers as input parameters and returns their least common multiple.

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

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



(a)

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

(b)

```
      1
    2 1 2
  3 2 1 2 3
4 3 2 1 2 3 4
```

(c)

```
5 4 3 2 1
4 3 2 1
3 2 1
2 1
1
```

(d)

```
1
2 2
3 3 3
4 4 4 4
5 5 5 5 5
```

(e)

```
1 2 3 4 5
  2 3 4 5
    3 4 5
      4 5
        5
```

(f)

```
* * * * *
*       *
*       *
*       *
* * * * *
```

(g)

```
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
```

(h)

```
      *
    * * *
  * * * * *
* * * * * * *
```

(i)

```
* * * * * * *
  *       *
    *   *
      *
```

(j)

```
* * * * * * *
  * * * * *
    * * *
      *
```

(k)

```
      *
    *   *
  *       *
*           *
  *       *
    *   *
      *
```

(l)

```
      *
    * * *
  * * * * *
* * * * * * *
  * * * * *
    * * *
      *
```

(m)

```
$ $ $ $ $
  $ $ $ $
    $ $ $
      $ $
        $
```

(n)

```
        #
      # #
    # # #
  # # # #
# # # # #
```

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

    a.

$$1 - x^2/2! + x^4/4! - x^6/6! + ...x^n/n!$$

    b.

$$e^x = 1 + x/1! + x^2/2! + x^3/3! + ...$$

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

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

16. 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 cubes of the digits is equal to the number itself. For Example,

$$370 = 3^3 + 7^3 + 0^3$$

17. 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 one.

18. Write a function to multiply two non-negative numbers by repeated addition, for example, 7*5 = 7+7+7+7+7.

19. Write a program that computes the average of a collection of values entered by the user. The user will enter 0 as a sentinel value to indicate that no further values will be provided. The program should display an appropriate error message if the first value entered by the user is 0

20. A particular retailer is having a 60 percent off sale on a variety of discontinued products. The retailer would like to help its customers determine the reduced price of the merchandise by having a printed discount table on the shelf that shows the original prices and the prices after the discount has been applied. Write a program that uses a loop to generate this table, showing the original price, the discount amount, and the new price for purchases of $4.95, $9.95, $14.95, $19.95 and $24.95. Ensure that the discount amounts and the new prices are rounded to 2 decimal places when displayed.

21. Write a program that reads an integer from the user. If the value entered by the user is less than 2 then your program should display an appropriate error message. Otherwise, your program should display the prime numbers that can be multiplied together to compute n, with one factor appearing on each line. For example:

```
Enter an integer (2 or greater): 72
The prime factors of 72 are:
2
2
2
3
3
```

Programming in Python (CSE 3142)
MINOR ASSIGNMENT-4: DEBUGGING

1. Consider the following Python code intended to compute the sum of n natural numbers. During testing, it was found that sum printed by program always excludes the last number. Debug the following script using the debugger.
   Program to compute the sum of n natural numbers

```python
def summation(n):
    '''
    Objective: To find sum of first n positive integers
    Input Parameter: n - numeric value
    Return Value: total - numeric value
    '''
    total = 0
    for count in range(1,n):
        total += count
    return total

def main():
    '''
    Objective: To find sum of first n positive integers based on user input
    Input Parameter: None
    Return Value: None
    '''
    n = int(input("Enter number of terms: "))
    total = summation(n)
    print("Sum of first",n,"positive integers:",total)

if __name__ == '__main__':
    main()
```

2. Consider the following Python code intended to print inverse right triangle for given numbers of rows nRows. For example, for nRows = 5, the following inverted triangle should be printed:

```
*****
****
***
**
*
```

   During testing, it was found that the program does not produce even the single line of output. Debug the following script using the debugger.
   Program to print inverse right triangle

```python
def invertedRightTriangle(nRows):
    '''
    Objective: To print right triangle
    Input Parameter: nRows - integer value
    Return Value: None
    '''
    for i in range(nRows,0):
        print("*"*i)

def main():
    '''
    Objective: To print right triangle
    Input Parameter: None
    Return Value: None
    '''
    nRows = int(input("Enter number of rows: "))
    invertedRightTriangle(nRows)

if __name__ == '__main__':
    main()
```

3. Consider the Python script given below intended to compute the percentage. During testing, it was found that percentage computed was not accurate rather rounded to lower bound integer value. Debug the following script using the debugger.
   Program to print inverse right triangle

```python
def main():
    '''
    Objective: To display percentage of marks scored by the student
    Input Parameter: None
    Return Value: None
    '''
    totalMarks = 0
    i = 0
    while True:
        marks = input("Enter marks for subject "+str(i+1)+":")
        if marks == '':
            break
        marks = int(marks)
        if marks<0 or marks>100:
            print("Invalid marks")
            continue
        i += 1
        totalMarks += marks
    percentage = totalMarks//i
    print("Total marks",int(totalMarks))
    print("Percentage",round(percentage,2))

if __name__ == '__main__':
    main()
```

4. Consider the Python given below intended to determine whether the given year is a leap year. During testing, it was found that an year such as 1800 or 2100, despite being non-leap year, was also displayed as a leap-year. Debug the following script using the debugger.
   Program to print inverse right triangle

```python
def isLeapYear(year):
    '''
    Objective: To determine whether a given year is a leap year or not
    Input Paramter: year - numeric value
    Return value: True if year us a leap year, False otherwise
    '''
    return year%400 == 0 or year%100 == 0 and year%4 == 0
```

5. Consider the Python script given below intended to find HCF. During testing, it was found that program yields an error for numbers having no common factor other than 1. Debug the following script using the debugger.
   Program to print inverse right triangle

```python
def findHCF(num1, num2):
    '''
    Objective: To find HCF of two numbers, num1 and num2,
    Input Parameters: num1, num2 - numeric values
    Return Value: HCF - numeric value
    '''
    if num1<num2:
        minNum = num1
    else:
        minNum = num2
    for i in range(minNum,1,-1):
        if num1%i == 0 and num2%i == 0:
            HCF = i
    return HCF

def main():
    '''
    Objective: To take two numbers as an input and find their HCF
    Input Parameter: None
    Return Value: None
    '''
    num1 = int(input("Enter first number: "))
    num2 = int(input("Enter second number: "))
    print(findHCF(num1, num2))

if __name__ == '__main__':
    main()
```

Programming in Python (CSE 3142)
MINOR ASSIGNMENT-5: SCOPE

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

i.
```
globalVar = 10
def test():
    localVar = 20
    print(’Inside function test: globalVar =’, globalVar)
    print(’Inside function test: localVar =’, localVar)
test()
print(’Outside function test: globalVar =’, globalVar)
print(’Outside function test: localVar =’, localVar)
```

ii.
```
globalVar = 10
def test():
    localVar = 20
    globalVar = 30
    print(’Inside function test: globalVar =’, globalVar)
    print(’Inside function test: localVar =’, localVar)
test()
print(’Outside function test: globalVar =’, globalVar)
```

iii.
```
globalVar = 10
def test():
    global globalVar
    localVar = 20
    globalVar = 30
    print(’Inside function test: globalVar =’, globalVar)
    print(’Inside function test: localVar =’, localVar)
test()
print(’Outside function test: globalVar =’, globalVar)
```

iv.
```
def test1():
    test1.a = 10
    def test2():
        test1.a = 8
        print(’Inside function test2 ’, test1.a)
    test2()
    print(’Outside function test2 ’, test1.a)
    test1()
```

v.
```
a = 4
def f():
    a = 5
    def g():
        nonlocal a
        a = 10
        print(’Inside function g, ’, ’a=’, a)
        def h():
            nonlocal a
            a = 20
            print(’Inside function h, ’, ’a=’, a)
        h()
        g()
        print(’Inside function f, ’, ’a=’, a)
    f()
vi. x = 2
    def test():
        x = x+1
        print(x)
    print(x)
vii. x = 2
    def test():
        global x
        x = x+1
        print(x)
    print(x)
```

Programming in Python (CSE 3142)

MINOR ASSIGNMENT-6: STRINGS

1. Write a function that takes a string as a parameter and returns a string with every successive repetitive character replaced with a star(*). For example, 'balloon' is returned as 'bal*o*n'.

2. Write a function that takes two strings and returns True if they are anagrams and False otherwise. A pair of strings is anagrams if the letters in one word can be arranged to form the second one.

3. Write a function that takes a sentence as an input parameter and displays the number of words in the sentence.

4. Write a function that takes a sentence as an input parameter and replaces the first letter of every word with the corresponding uppercase letter and rest of the letters in the word by corresponding letters in lowercase without using builtin function.

5. Write a function that takes a string as an input and determines the count of the number of words using regular expression.

6. What will be the output on executing each of the statements, following the assignment statement:
   address = 'B-6, Lodhi road, Delhi'
   a. len(address)
   b. address[17:-1]
   c. address[-len(address): len(address)]
   d. address[:-12] + address[-12:]
   e. address.find('delhi')
   f. address.swapcase()
   g. address.split(',')
   h. address.isalpha()

7. Examine the following string:
   greeting = 'Good Morning. Have a Good Day!!'
   What will be the output for the following function calls:
   a. greeting.count('Good')
   b. greeting.find('a')
   c. greeting.rfind('a')
   d. greeting.capitalize()
   e. greeting.lower()
   f. greeting.upper()
   g. greeting.swapcase()
   h. greeting.istitle()
   i. greeting.replace('Good', 'Sweet')
   j. greeting.strip()
   k. greeting.split()
   l. greeting.partition('.')

m. greeting.startswith(’good’)
n. greeting.endswith(’!!’)

8. Determine the patterns extracted by the following regular expressions.
1. string1 = ’Python Programming Language’

1. match1 = re.search(’......m?’, string1)
print(match1.group())
2. match2 = re.search(’......m{1,2}’, string1)
print(match2.group())
3. match3 = re.search(’.*Language$’, string1)
print(match3.group())
4. match4 = re.search(’\w*\s\w*’, string1)
print(match4.group())
5. match5 = re.search(’.*’, string1)
print(match5.group())

2. string2 = ’Car Number DL5645’

1. match1 = re.search(’\w\w?\d{1,4}’, string2)
print(match1.group())
2. match2 = re.search(’.*5’, string2)
print(match2.group())
3. match3 = re.search(’.*5?’, string2)
print(match3.group())
4. match4 = re.search(’\d{3}’, string2)
print(match4.group())
5. match5 = re.search(’^ C.*5$’, string2)
print(match5.group())

3. string3 = ’cdcccdcddd343344aabb’
1. match1 = re.search(’(c|d)*\d*(a|b)*’, string3)
print(match1.group())
2. match2 = re.search(’(cd)*d’, string3)
print(match2.group())
3. match3 = re.search(’(cc|cd)*(3|4)*(aa|bb)’, string3)
print(match3.group())
4. match4 = re.search(’(cc|cd|dd)*(3|4)*(aa|bb)’, string3)
print(match4.group())
5. match5 = re.search(’(cc|cd|dd)*(3|4)*(aa|bb)*’, string3)
print(match5.group())

9. Write a python programs to perform the following tasks:
a. Reverse a string
b. Reverse a string without reversing the words. Example:
input data: ’welcome to iter’
output: ’iter to welcome’

c. Check if a string is symmetric or asymmetric
d. Check if a string is palindrome.
e. Given a string s and index i, delete ith value from s
f. Count the number of vowels and consonants in a string.
g. Find length of a string without using inbuilt function.
h. Check if a string contains at least one digit and one alphabet.
i. Remove duplicates from a string.
j. Count frequency of characters in a string.
k. Find the character having maximum frequency in a string.

Programming in Python (CSE 3142)

MINOR ASSIGNMENT-7: MUTABLE AND IMMUTABLE OBJECTS

- Write a function that takes a list of values as input parameter and returns another list without any duplicates.

- Write a function that takes a list of numbers as input from the user and produces the corresponding cumulative list where each element in the list at index i is the sum of elements at index j <= i.

- Write a program that takes a sentence as input from the user and computes the frequency of each letter. Use a variable of dictionary type to maintain the count.

- Identify the output produced when the following functions are invoked.

1.
```
def func():
    l1 = list()
    l2 = list()
    for i in range(0,5):
        l1.append(i)
        l2.append(i+3)
    print(l1)
    print(l2)
```

2.
```
def func():
    l1 = list()
    l2 = list()
    for i in range(0,5):
        l1.append(i)
        l2.append(i+3)
        l1, l2 = l2, l1
    print(l1)
    print(l2)
```

- Determine the output of the following code snippets:

1.
```
c = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
result = 0
for i in range(0, 10):
    if (c[i]%2 == 0):
        result += c[i]
print(result)
```

2.
```
c = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
result = 0
for i in range(0, 10):
    if (c[i]%2 != 0):
        result += c[i]
print(result)
```

3.
   ```python
   subject = 'computer'
   subject = list(subject)
   ch = subject[0]
   for i in range(0, len(subject)-1):
       subject[i] = subject[i+1]
   subject[len(subject)-1]=ch
   print(''.join(subject))
   ```

4.
   ```python
   quantity = [15, 30, 12, 34, 56, 99]
   total = 0
   for i in range(0, len(quantity)):
       if (quantity[i] > 15):
           total += quantity[i]
   print(total)
   ```

5.
   ```python
   x = [1, 2, 4, 6, 9, 10, 14, 15, 17]
   for i in range(0, len(x)):
       if (x[i]%2 == 0):
           x[i] = 4*i
       elif (x[i]%3 == 0):
           x[i] = 9*i
       else:
           x[i] *= 2
   print(x)
   ```

- Write a function that takes **n** as an input and creates a list of **n** lists such that i^{th} list contains first five multiples of **i**.

- Write a function that takes a number as an input parameter and returns the correspond text in words, for example, on input 452, the function should return ‘Four Five Two’. Use a dictionary for mapping digits to their string representation.

- Given the following inputs, indicate in each case (a) to (w), whether the statements will execute successfully. If, so, give what will be the outcome of execution? Also give the output of print statements (where applicable):

```python
address = 'B-6, Lodhi road, Delhi'
list1 = [1, 2, 3]
list2 = ['a', 1, 'z', 26, 'd', 4]
tuple1 = ('a', 'e', 'i', 'o', 'u')
tuple2 = ([2,4,6,8], [3,6,9], [4,8], 5)
dict1 = {'apple': 'red', 'mango': 'yellow', 'orange': 'orange'}
dict2 = {'X': ['eng', 'hindi', 'maths','science'], 'XII': ['english', 'physics',
    'chemistry', 'maths']}
```

1.
   ```python
   list1[3] = 4
   ```

2.
   ```python
   print(list1 * 2)
   ```

3.
   ```
   print(min(list2))
   ```

4.
   ```
   print(max(list1))
   ```

5.
   ```
   print(list(address))
   ```

6.
   ```
   list2.extend([’e’, 5])
   print(list2)
   ```

7.
   ```
   list2.append([’e’, 5])
   print(list2)
   ```

8.
   ```
   names = [’rohan’, ’mohan’, ’gita’]
   names.sort(key= len)
   print(names)
   ```

9.
   ```
   list3 = [(x * 2) for x in range(1, 11)]
   print(list3)
   ```

10.
   ```
   del list3[1:]
   print(list3)
   ```

11.
   ```
   list4 = [ x+y for x in range(1,5) for y in range(1,5)]
   print(list4)
   ```

12.
   ```
   tuple2[3] = 6
   ```

13.
   ```
   tuple2.append(5)
   ```

14.
   ```
   t1 = tuple2 +(5)
   ```

15.
   ```
   ’,’.join(tuple1)
   ```

16.
   ```
   list(zip([’apple’, ’orange’], (’red’,’orange’)))
   ```

17.
   ```
   dict2[’XII’]
   ```

18. `dict2[’XII’].append(’computer science’),dict2`

19. `’red’ in dict1`

20. `list(dict1.items())`

21. `list(dict2.keys())`

22. `dict2.get(’XI’, ’None’)`

23.
    ```
    dict1.update({’kiwi’:’green’})
    print(dict1)
    ```

- Consider the following three sets, namely vehicles, heavyVehicles, and lightVehicles:
  ⟩⟩⟩ vehicles = {‘Bicycle’, ‘Scooter’, ‘Car’, ‘Bike’, ‘Truck’, ‘Bus’, ‘Rickshaw’}
  ⟩⟩⟩ heavyVehicles = {‘Truck’, ‘Bus’}
  ⟩⟩⟩ lightVehicles = {‘Rickshaw’, ‘Scooter’, ‘Bike’, ‘Bicycle’}
  Determine the output on executing the following statements:

  1.
     ```
     lytVehicles = vehicles - heavyVehicles
     print(lytVehicles)
     ```

  2.
     ```
     hvyVehicles = vehicles - lightVehicles
     print(hvyVehicles)
     ```

  3.
     ```
     averageWeightVehicles = lytVehicles & hvyVehicles
     print(averageWeightVehicles)
     ```

  4.
     ```
     transport = lightVehicles | heavyVehicles
     print(transport)
     ```

  5.
     ```
     transport.add(’Car’)
     print(transport)
     ```

  6.
     ```
     for i in vehicles:
         print(i)
     ```

  7. `len(vehicles)`

  8. `min(vehicles)`

  9. `set.union(vehicles, lightVehicles, heavyVehicles)`