**Day 1 Assignments**

**Question 1:**

1. Write a Python program to accept user input (with input() function) of integer (binary, octal, decimal

and Hexadecimal), float (using decimal or exponent formats, e and E) and strings (all 3 ways) and

print the user entered values.

4. Perform the following with bit-wise operators (check manually with system generated results)

 v1 = 45, v2 = 56, perform v1 & v2, v1 | v2, v1 ^ v2, ~v1, ~v2, v1 << 1, v1 >> 2

 v1 = 034, v2 = 045, perform v1 & v2, v1 | v2, v1 ^ v2, ~v1, ~v2, v1 << 2, v1 >> 3

 v1 = 0xAB, v2 = 0x89, perform v1 & v2, v1 | v2, v1 ^ v2, ~v1, ~v2, v1 << 2, v1 >> 3

 v1 = 0b11011100, v2 = 0b01101010, perform v1 & v2, v1 | v2, v1 ^ v2, ~v1, ~v2, v1 << 2, v1 >> 3

#!/usr/bin/python

de = input("Enter Decimal Value : ")

print int(de)

bi = input("Enter Binary Value : ")

print bin(bi)

oc = input("Enter Octal Value : ")

print oct(oc)

he = input("Enter Hexadecimal Value : ")

print hex(he)

fl = input("Enter float Value : ")

print float(fl)

sq = input("Enter String with Single Quote : ")

print sq

dq = input("Enter String with Double Quote : ")

print dq

tq = input("Enter String with Triple Quote : ")

print tq

**Answer:**

gadikgop@VTA077l:~/python/day1$ ./first.py

Enter Decimal Value : 901

901

Enter Binary Value : 0b1010101

0b1010101

Enter Octal Value : 016

016

Enter Hexadecimal Value : 0x1DF

0x1df

Enter float Value : 12.4

12.4

Enter String with Single Quote : 'hai hello'

hai hello

Enter String with Double Quote : "hai hello"

hai hello

Enter String with Triple Quote : """hai hello"""

hai hello

**Question 2:**

2. Write a Python program to perform the following:

 Assign integer (decimal) value to an variable and print the variable

 Assign integer (binary) value to an variable and print the variable

 Assign integer (octal) value to an variable and print the variable

 Assign integer (hexa-decimal) value to an variable and print the variable

#!/usr/bin/python

decimal=90

binary=0b1010011

octal=067

hexa=0x23

print decimal

print binary

print octal

print hexa

**Answer:**

gadikgop@VTA077l:~/python/day1$ ./second.py

90

83

55

35

**Question 3:**

3. Write the steps in Python to perform the following data type conversions:

 Assign values in binary, octal, hexadecimal, decimal, print the values using int() function with

explicit base. Also print the type of the values.

 Assign values in binary, octal, hexadecimal, decimal, print the values using long() function with

explicit base. Also print the type of the values.

 Accept an integer from user, convert to float and print the number.

 Accept 3 integers, convert them to strings, add the 3 strings and display the resultant string

 Accept a string from user, print char and its integer value for all the characters (No need to use

loops)

 Accept an integer (between 65 and 90), print its equivalent character (No need of loops or

conditional statements)

 Accept an integer (decimal) from user, print in binary, octal, decimal and hexadecimal formats

 Accept an integer (binary) from user, print in binary, octal, decimal and hexadecimal formats

 Accept an integer (octal) from user, print in binary, octal, decimal and hexadecimal formats

 Accept an integer (hexadecimal) from user, print in binary, octal, decimal and hexadecimal

formats

 Accept an integer from user, pass to octal function and print

 Accept an integer from user, convert to binary, pass to octal function and print

 Accept an integer from user, pass to hexadecimal function and print

 Accept an integer from user, convert to binary, pass to hexadecimal function and print

 Accept an integer from user, convert to octal, pass to hexadecimal function and print

#!/usr/bin/python

decimal="712"

binary="0b1010"

octal="0712"

hexa="0x712"

print "First..."

d=int(decimal,0)

print "Decimal Value is",d,"type is",type(d)

b=int(binary,2)

print "Binary Value is",b,"type is",type(b)

o=int(octal,8)

print "Octal Value is",o,"type is",type(o)

h=int(hexa,16)

print "Hexa Decimal Value is",h,"type is",type(h)

print

print "Second...."

print

d=long(decimal,0)

print "Decimal Value is",d,"type is",type(d)

b=long(binary,2)

print "Binary Value is",b,"type is",type(b)

o=long(octal,8)

print "Octal Value is",o,"type is",type(o)

h=long(hexa,16)

print "Hexa Decimal Value is",h,"type is",type(h)

print

print "Third....."

val = input("Enter a Value : ")

print float(val)

print

print "Fourth....."

val1 = str(input("Enter a Value 1 : "))

val2 = str(input("Enter a Value 2 : "))

val3 = str(input("Enter a Value 3 : "))

res = val1+val2+val3

print "Resultant String is",res

print

print "Fifth..."

st = raw\_input("Enter String of length 6 : ")

print st[0],"-",ord(st[0]),"\n"

print st[1],"-",ord(st[1]),"\n"

print st[2],"-",ord(st[2]),"\n"

print st[3],"-",ord(st[3]),"\n"

print st[4],"-",ord(st[4]),"\n"

print st[5],"-",ord(st[5])

print

print "Sixth....."

num = input("Enter a Values in b/w 65 to 90 : ")

print "Character of the value",chr(num)

print

print "Seventh...."

val = input("Enter Integer : ")

print "Integer Value is",(val)

print "Integer - Binary Value is",bin(val)

print "Integer - Octal Value is",oct(val)

print "Integer - Decimal Value is",float(val)

print "Integer - Hexa Decimal Value is",hex(val)

print

print "Eighth...."

val = bin(input("Enter Binary Value : "))

print "Binary Value is",(val)

print "Binary - Integer Value is",int(val,2)

print "Binary - Octal Value is",oct(int(val,2))

print "Binary - Decimal Value is",float(int(val,2))

print "Binary - Hexa Decimal Value is",hex(int(val,2))

print

print "Nine..."

val = oct(input("Enter Octal Value : "))

print "Octal Value is",(val)

print "Octal - Integer Value is",int(val,8)

print "Octal - Bin Value is",bin(int(val,8))

print "Octal - Decimal Value is",float(int(val,8))

print "Octal - Hexa Decimal Value is",hex(int(val,8))

print

print "Ten...."

val = hex(input("Enter Hexa Decimal Value : "))

print "Hexa Decimal Value is",(val)

print "Hexa Decimal - Integer Value is",int(val,16)

print "Hexa Decimal - Bin Value is",bin(int(val,16))

print "Hexa Decimal - Decimal Value is",float(int(val,16))

print "Hexa Decimal - Octal Value is",oct(int(val,16))

**Answer:**

gadikgop@VTA077l:~/python/day1$ ./third.py

First...

Decimal Value is 712 type is <type 'int'>

Binary Value is 10 type is <type 'int'>

Octal Value is 458 type is <type 'int'>

Hexa Decimal Value is 1810 type is <type 'int'>

Second....

Decimal Value is 712 type is <type 'long'>

Binary Value is 10 type is <type 'long'>

Octal Value is 458 type is <type 'long'>

Hexa Decimal Value is 1810 type is <type 'long'>

Third.....

Enter a Value : 123

123.0

Fourth.....

Enter a Value 1 : 12

Enter a Value 2 : 21

Enter a Value 3 : 21

Resultant String is 122121

Fifth...

Enter String of length 6 : sdlkdd

s - 115

d - 100

l - 108

k - 107

d - 100

d - 100

Sixth.....

Enter a Values in b/w 65 to 90 : 78

Character of the value N

Seventh....

Enter Integer : 23

Integer Value is 23

Integer - Binary Value is 0b10111

Integer - Octal Value is 027

Integer - Decimal Value is 23.0

Integer - Hexa Decimal Value is 0x17

Eight....

Enter Binary Value : 0b100101

Binary Value is 0b100101

Binary - Integer Value is 37

Binary - Octal Value is 045

Binary - Decimal Value is 37.0

Binary - Hexa Decimal Value is 0x25

Nine...

Enter Octal Value : 0123

Octal Value is 0123

Octal - Integer Value is 83

Octal - Bin Value is 0b1010011

Octal - Decimal Value is 83.0

Octal - Hexa Decimal Value is 0x53

Ten....

Enter Hexa Decimal Value : 0x2BA

Hexa Decimal Value is 0x2ba

Hexa Decimal - Integer Value is 698

Hexa Decimal - Bin Value is 0b1010111010

Hexa Decimal - Decimal Value is 698.0

Hexa Decimal - Octal Value is 01272

**Question 4:**

#!/usr/bin/python

v1 = 45

v2 = 56

print v1 & v2, v1 | v2, v1 ^ v2, ~v1, ~v2, v1 << 1, v1 >> 2

v1 = 0xAB

v2 = 0x89

print v1 & v2, v1 | v2, v1 ^ v2, ~v1, ~v2, v1 << 2, v1 >> 3

v1 = 0b11011100

v2 = 0b01101010

print v1 & v2, v1 | v2, v1 ^ v2, ~v1, ~v2, v1 << 2, v1 >> 3

**Answer:**

gadikgop@VTA077l:~/python/day1$ ./four.py

40 61 21 -46 -57 90 11

137 171 34 -172 -138 684 21

72 254 182 -221 -107 880 27

**Question 5:**

5. Write a program to perform the setting and clearing bits of 2 nd and 4 th respectively of variable v1,

say, v1 = 0xCD, check the results

Positions

v1 = 0XCD

#!/usr/bin/python

v1=0XCD

print bin(v1)

n=input("Enter the position you want to set :")

var=0x1<<n

d=v1 | var

print "Output after setting the value ",bin(d)

n1=input("Enter the input you want to clear :")

var1=~(0x01<<n1)

d=v1 & var1

print "Output after clearing the value :",bin(d)

**Answer:**

gadikgop@VTA077l:~/python/day1$ ./five.py

0b11001101

Enter the position you want to set :5

Output after setting the value 0b11101101

Enter the input you want to clear :3

Output after clearing the value : 0b11000101

**Day 2 Assignments**

**Question 1**

1. Write a program which accepts a sequence of comma-separated numbers from console and

generate a list and a tuple which contains every number.

Input: 10, 20, 30, 50, 40

Output:

List: ['10', '20', '30', '50', '40']

Tuple: ('10', '20', '30', '50', '40')

#!/usr/bin/python

inp=input("Enter the input numbers :")

l=[]

for val in inp:

l.append(val)

print l

print tuple(l)

**Answer:**

gadikgop@VTA077l:~/python/day2$ ./first.py

Enter the input numbers :10,20,30,40

[10, 20, 30, 40]

(10, 20, 30, 40)

**Question 2**

2. Write a program which will find all the numbers which are divisible by 4 but are not a multiple of 5

in between user provided range. The numbers obtained should be printed in a comma-separated

sequence on a single line.

#!/usr/bin/python

inp1=input("Enter the number1 :")

inp2=input("Enter the number2 :")

l=[]

for val in range(inp1,inp2):

if (val%4==0) and (val%5!=0):

l.append(str(val))

print ",".join(l)

**Answer:**

gadikgop@VTA077l:~/python/day2$ ./second.py

Enter the number1 :10

Enter the number2 :200

12,16,24,28,32,36,44,48,52,56,64,68,72,76,84,88,92,96,104,108,112,116,124,128,132,136,144,148,152,156,164,168,172,176,184,188,192,196

**Question 3**

3. With a user given integral number num, write a program to generate a dictionary that contains

(num, num \* num \* num) such that is an integral number between 1 and num (both included) and

print the dictionary.

Input: 5

Output: {1: 1, 2: 8, 3: 27, 4: 64, 5: 125}

#!/usr/bin/python

num=input("Enter the number :")

d=dict()

for val in range(1,num+1):

d[val]=val\*val\*val

print d

**Answer:**

gadikgop@VTA077l:~/python/day2$ ./third.py

Enter the number :10

{1: 1, 2: 8, 3: 27, 4: 64, 5: 125, 6: 216, 7: 343, 8: 512, 9: 729, 10: 1000}

**Question 4**

4. Write a program to print index numbers along with values from the list, l1=[2, 13, 5, 7, 6, 9], where

two elements sums to 15

#!/usr/bin/python

l1=[2, 13, 5, 7, 6, 9]

for val in range(len(l1)):

for val1 in range(val):

if (l1[val]+l1[val1]==15):

print "Values ",l1[val1],l1[val],"Indexes ",val1,val

**Answer**

gadikgop@VTA077l:~/python/day2$ ./four.py

Values 2 13 Indexes 0 1

Values 6 9 Indexes 4 5

**Question 5**

5. Write a program, which will find all such numbers between 1000 and 3000 (both included) so that

each digit of the number is an even number. The numbers obtained should be printed in a comma-

separated sequence on a single line.

#!/usr/bin/python

values = []

for i in range(1000, 3001):

s = str(i)

if (int(s[0])%2==0) and (int(s[1])%2==0) and (int(s[2])%2==0) and (int(s[3])%2==0):

values.append(s)

print ",".join(values)

**Answer:**

gadikgop@VTA077l:~/python/day2$ ./five.py

2000,2002,2004,2006,2008,2020,2022,2024,2026,2028,2040,2042,2044,2046,2048,2060,2062,2064,2066,2068,2080,2082,2084,2086,2088,2200,2202,2204,2206,2208,2220,2222,2224,2226,2228,2240,2242,2244,2246,2248,2260,2262,2264,2266,2268,2280,2282,2284,2286,2288,2400,2402,2404,2406,2408,2420,2422,2424,2426,2428,2440,2442,2444,2446,2448,2460,2462,2464,2466,2468,2480,2482,2484,2486,2488,2600,2602,2604,2606,2608,2620,2622,2624,2626,2628,2640,2642,2644,2646,2648,2660,2662,2664,2666,2668,2680,2682,2684,2686,2688,2800,2802,2804,2806,2808,2820,2822,2824,2826,2828,2840,2842,2844,2846,2848,2860,2862,2864,2866,2868,2880,2882,2884,2886,2888

**Day 3**

**Question 1:**

1. Write a program that returns a list containing only the elements that are common for the two lists.

#!/usr/bin/python

lst1=input("enter a list: ")

lst2=input("enter a list: ")

l=[]

for val in lst1:

if val in lst2:

l.append(val)

print l

**Answer:**

gadikgop@VTA077ll:~/Desktop/python/11.py$ ./first.py

enter a list: [1,2,3,4,5,6,7,8]

enter a list: [4,5,6,7,8,9]

[4, 5, 6, 7, 8]

**Question 2:**

2. Write a program in Python to perform the following:

• Accept user input of +ve and –ve values and store it in list, say mylist (To end user input pass 0)

• Move all negative elements to end.

Input : mylist = [1, -1, 3, 2, -7, -5, 11, 6 ]

Output : [1 3 2 11 6 -1 -7 -5]

Input : mylist = [-5, 7, -3, -4, 9, 10, -1, 11]

Output : [7 9 10 11 -5 -3 -4 -1]

#!/usr/bin/python

n=input("enter num: ")

mylist=[]

while n!=0:

mylist.append(n)

n=input("enter num: ")

print mylist

l1=[]

l2=[]

for i in mylist:

if (i>=0):

l1.append(i)

if i<0:

l2.append(i)

l=l1+l2

print l

**Answer:**

gadikgop@VTA077ll:~/Desktop/python/11.py$ ./11\_12\_2.py

enter num: 2

enter num: 4

enter num: -5

enter num: -6

enter num: 8

enter num: 9

enter num: 0

[2, 4, -5, -6, 8, 9]

[2, 4, 8, 9, -5, -6]

**Question 3:**

3. Write a function to receive start and end numbers, calculate prime numbers between start and end,

return the primes in list to the caller. If start and end are not passed take default as 1, 100.

#!/usr/bin/python

def primei(start=1,end=100):

lst=[]

for val in range(start,end+1):

count=0

for i in range(2,(val/2)+1):

if(val%i==0):

count=count+1

if(count<=0):

lst.append(val)

return lst

a=primei()

b=primei(4,20)

print a

print b

**Answer:**

gadikgop@VTA077ll:~/Desktop/python/11.py$ ./11\_12\_3.py

[1, 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97]

[5, 7, 11, 13, 17, 19]

**Question 4**

4. Write a function to receive list and number, then it has to multiply each element of the list with

number. Print the new list in function and in main. (Two methods, create new list, modify existing

list).

#!/usr/bin/python

def func1(lst,num):

new\_list=[]

for i in lst:

a=i\*num

new\_list.append(a)

print "fun1: ",new\_list

return new\_list

if \_\_name\_\_ == "\_\_main\_\_":

lst=[1,2,3,4]

num=10

print func1(lst,num)

def func2(lst, num):

for index in range(len(lst)):

lst[index] = lst[index]\*num

print "fun2: ",lst

return lst

if \_\_name\_\_ == "\_\_main\_\_":

lst = [2,6,5,4]

num = 8

print func2(lst, num)

**Answer:**

gadikgop@VTA077ll:~/Desktop/python/11.py$ ./ans4.py

fun1: [10, 20, 30, 40]

[10, 20, 30, 40]

fun2: [16, 48, 40, 32]

[16, 48, 40, 32]

**Question 5:**

5. Write a function to receive variable arguments and need to perform sum of the variable arguments

and return the result. Print the result in main program.

#!/usr/bin/python

def func(\*nums):

su=0

for i in nums:

su=su+i

return su

if \_\_name\_\_ == "\_\_main\_\_":

print(func(2,4,5,6))

def func(\*nums):

return sum(nums)

if \_\_name\_\_ == "\_\_main\_\_":

print(func(2,4,5,6))

**Answer:**

gadikgop@VTA077ll:~/Desktop/python/11.py$ ./ans5.py

17

**Day 4 Assignments**

**Question 1:**

1. Write a program to accept two complex numbers from user and perform complex numbers

operations (addition, subtraction and multiplication of complex numbers)

#!/usr/bin/python

val1 = input("Enter Complex Val 1 :")

val2 = input("Enter Complex Val 2 :")

c\_sum = val1 + val2

c\_sub = val1 - val2

c\_pro = val1 \* val2

print "Sum is :",c\_sum

print "Difference is :",c\_sub

print "Product is :",c\_pro

**Answer**

gadikgop@VTA077l:~/Downloads/12-DEC-2017$ python 1\_Complex\_Arithmetic.py

Enter Complex Val 1 :3

Enter Complex Val 2 :4

Sum is : 7

Difference is : -1

Product is : 12

**Question 2**

2. Write a program which takes 2 digits, v1, v2 as user input and generates/displays a 2-dimensional

array. The element value in the i-th row and j-th column of the array should be i\*j

Note: Ranges of i and j are 0,1, ..., v1 - 1 and 0, 1, ..., v2-1 respectively

#!/usr/bin/python

val1 = input("Enter val1 :")

val2 = input("Enter val2: ")

for i in range(val1-1):

for j in range(val2-1):

mul = i\*j

print mul,

print

**Asnwer**

gadikgop@VTA077l:~/Downloads/12-DEC-2017$ python 2\_2D\_Array.py

Enter val1 :5

Enter val2: 6

0 0 0 0 0

0 1 2 3 4

0 2 4 6 8

0 3 6 9 12

**Question 3:**

3. Write a program in Python to find smallest factorial number containing given trailing zeros. Given a

number num. The task is to find the smallest number whose factorial contains at least num trailing

zeroes.

#!/usr/bin/python

num = input("Enter a value :")

mul = 1;cnt = 0

for val in range(1,100):

mul = mul \* val

pro = mul

cnt = 0

while(pro !=0 ):

ld = pro % 10

if ld == 0:

cnt = cnt + 1

else:

break

pro = pro / 10

if cnt == num:

print val,"!","is trailing with",num,"zeros"

break

**Answer:**

gadikgop@VTA077l:~/Downloads/12-DEC-2017$ python 3\_Trailing\_Factorial.py

Enter a value :4

20 ! is trailing with 4 zeros

**Question 4:**

4. Write a program in Python to implement the below functionalities using map, filter and reduce

functions:

 Generate a random number between 10 and 20

 Using above generated random number build a list within range of -10 to 20

 Print the above list

 Process the numbers in the above list and retrieve only positive numbers, store in list and print

them

 Using above positive numbers find squares of numbers, store in list and print them

 Using above square of numbers find sum of squares of numbers, store in list and print them

#!/usr/bin/python

import random

lst = []

for x in range(1):

val = random.randint(-10,20)

print "Randon Number in b/w: -10 to 20 : ",val

for i in range(-10,val):

lst.append(i)

print "List based on random Number : ",lst

lst\_p = filter(lambda x:x>10,lst)

print "Positive Numbers : ",lst\_p

lst\_sq = map(lambda x:x\*x,lst\_p)

print "Square of Positive Numbers : ",lst\_sq

sum\_sq = reduce(lambda v1,v2:v1+v2,lst\_sq)

print "Sum of Squares",sum\_sq

**Answer:**

gadikgop@VTA077l:~/python/python\_exam$ python 3.py

length of list : 25

(-15,15)random list : [-12, 9, 2, 3, -13, -3, -7, 6, -11, 14, 10, -5, -12, 14, 13, -2, -7, -4, 2, -12, -1, 8, -1, -10, 2]

positive numbers list : [9, 2, 3, 6, 14, 10, 14, 13, 2, 8, 2]

cubes of the list : [729, 8, 27, 216, 2744, 1000, 2744, 2197, 8, 512, 8]

sum of above list : 10193

**Question 5:**

5. Write a program in Python to perform the following:

 Accept a list from user

 Accept a tuple from user

 Form list comprehension of square of numbers using above list and print the result

 Form tuple comprehension of square of numbers using above tuple and print the result

 Form set comprehension of square of numbers using above list and print the result

 Accept a list from user. If user enters one number then accept that as stop, if user enters two

numbers then accept them as start and stop, if user enters 3 numbers then treat them as start,

stop and step. If user enters invalid start, stop and step numbers or more than 3 numbers then

take start as 0, step as 1 and stop as 10. Using start, stop and step form a list.

 Using above list and comprehensions, form list of even multiples and display them.

 Write sample code for nested comprehension.

 Accept 3 numbers from user. Add the numbers and display the list using lambda functions.

lst=list(input("enter list: "))

tpl=input("enter tuple: ")

lst\_sq=[x\*\*2 for x in lst]

print "list square: ",lst\_sq

tpl\_sq=tuple(x\*\*2 for x in tpl)

print "tuple square: ",tpl\_sq

set\_sq={x\*\*2 for x in lst}

print "set square: ",set\_sq

ls=[]

def sss\_values(start=0,stop=10,step=1):

for val in range(start,stop,step):

ls.append(val)

print "main list: ",ls

lst1=list(input("enter a list for start,stop,step: "))

if len(lst1)==1:

stop=lst1[0]

sss\_values(0,stop)

elif len(lst1)==2:

start=lst1[0]

stop=lst1[1]

sss\_values(start,stop)

elif len(lst1)==3:

start=lst1[0]

stop=lst1[1]

step=lst1[2]

sss\_values(start,stop,step)

else:

sss\_values()

even\_list=[x for x in ls if x%2==0]

print "Even numbrs list: ",even\_list

lst\_n=list(input("enter list in list as input: "))

nested\_lst=[y for x in lst\_n for y in x]

print nested\_lst

nums=list(input("enter 2 numbers as input: "))

add=reduce(lambda x,y:x+y,nums)

print "list: ",nums

print "addition of nums: ",add

**Answer:**

gadikgop@VTA077l:~/python/day4$ python 5\_2.py

enter list: 1,2,3,54

enter tuple: 2,1,324,2

list square: [1, 4, 9, 2916]

tuple square: (4, 1, 104976, 4)

set square: set([2916, 1, 4, 9])

enter a list for start,stop,step: 1,20,2

main list: [1, 3, 5, 7, 9, 11, 13, 15, 17, 19]

Even numbrs list: []

enter 2 numbers as input: 12,312

list: [12, 312]

addition of nums: 324

**Day 5 Assignments**

**Question 1:**

1. Write a program in Python to perform command line arguments multiplication. If no command line

arguments are passed print error message and print result if proper arguments are passed.

#!/usr/bin/python

import sys

l=len(sys.argv)

mul=1

if (l>1):

for args in range(1,l):

mul\*=int(sys.argv[args])

print "Multiplication of input numbers :",mul

else:

print "Please pass arguments.."

**Answer:**

gadikgop@VTA077l:~/python/day5$ ./first.py 2 3 4 5

Multiplication of input numbers : 120

gadikgop@VTA077l:~/python/day5$

**Question 2:**

2. Develop Python module to address following 3 functionalities of Fibonacci Series:

 Normal method - Function to print the required fibonacci series, if function receives 10, then

prints 10 fibonacci numbers

 List - Function would return the list of fibonacci series

 Tuple - Function would return the fibonacci numbers in a tuple

Include main program to use above modules, main program to print the numbers (not list or tuple

as it is)

#!/usr/bin/python

def fibonacci(inp):

summ=0

v1=0

v2=1

print v1,v2,

while summ<(inp-2):

val=v1+v2

v1=v2

v2=val

print val,

summ=summ+1

print

inp=input("Enter the input from the user :")

fibonacci(inp)

#!/usr/bin/python

def fib\_l(inp):

v1=0

v2=1

summ=0

l=[v1,v2]

while (summ<inp-2):

val=v1+v2

l.append(val)

v1=v2

v2=val

summ+=1

return l

inp=input("Enter the input number:")

print fib\_l(inp)

#!/usr/bin/python

def fib\_l(inp):

v1=0

v2=1

summ=0

l=[v1,v2]

while (summ<inp-2):

val=v1+v2

l.append(val)

v1=v2

v2=val

summ+=1

return tuple(l)

inp=input("Enter the input number:")

print fib\_l(inp)

#!/usr/bin/python

import fibo,fibo\_list,fibo\_tuple

fibo.fibonacci(10)

**Answers**

adikgop@VTA077l:~/python/day5/second$ ./main.py

Enter the input from the user :10

0 1 1 2 3 5 8 13 21 34

0 1 1 2 3 5 8 13 21 34

**Question 3:**

3. Develop Python module to address following 3 functionalities of Prime Numbers:

 Normal method - Function to print the whether passed number is prime or not, if function

receives 10, then prints message ""10 is not a prime number""

 List - Function would return the list of prime numbers accepting the range as arguments (if no

arguments passed then default, it should take 1 and 100 as range)

 Tuple - Function would return the list of prime numbers accepting the range as arguments (if no

arguments passed then default, it should take 50 and 200 as range)

Include main program to use above modules, main program to print the numbers (not list or tuple as it

is)

#!/usr/bin/python

def prime\_num(inp):

for val in range(2,(inp/2)+1):

if (inp%val==0):

print "%d is not a prime number"%inp

break

else:

print "%d is prime number"%inp

inp=input("Enter the input number :")

prime\_num(inp)

#!/usr/bin/python

def prime\_list(inp1=1,inp2=100):

l=[]

for val in range(inp1,inp2):

if(val>1):

for val1 in range(2,(val/2)+1):

if (val%val1==0):

break

else:

l.append(val)

return l

print prime\_list()

inp1=input("Enter the starting number :")

inp2=input("Enter the ending number :")

print prime\_list(inp1,inp2)

#!/usr/bin/python

def prime\_list(inp1=50,inp2=200):

l=[]

for val in range(inp1,inp2):

if(val>1):

for val1 in range(2,(val/2)+1):

if (val%val1==0):

break

else:

l.append(val)

return tuple(l)

print prime\_list()

inp1=input("Enter the starting number :")

inp2=input("Enter the ending number :")

print prime\_list(inp1,inp2)

#!/usr/bin/python

import prime

prime.prime\_num(20)

**Answer:**

gadikgop@VTA077l:~/python/day5/third$ ./main.py

Enter the input number :11

11 is prime number

20 is not a prime number

**Question 4:**

4. Develop Python module to address following 3 functionalities of Factorial of given numbers:

 Normal method - Function to print the factorial of passed number, if function receives 5, then

prints factorial of 5 is 120

 List - Function would return the list of factorial values of passed list, if function receives [3, 5, 4]

then it function to return [6, 120, 24]

 Tuple - Function would return the factorial values of passed tuple in a tuple

NOTE: In list you implement logic of multiplying in descending order as 5 \* 4 \* 3 \* 2 \* 1 = 120 and in

tuple implement logic of multiplying in ascending order as 1 \* 2 \* 3 \* 4 \* 5 = 120

Include main program to use above modules, main program to print the individual numbers (not list or

tuple as it is)"

#!/usr/bin/python

def factorial(inp):

val=1

for val1 in range(1,inp+1):

val\*=val1

return val

inp=input("Enter the number :")

print "Factorial of %d is"%(inp),factorial(inp)

#!/usr/bin/python

def factorial\_list(inp):

l=[]

for val1 in inp:

val=val1

for v in range(val1-1,1,-1):

val\*=v

l.append(val)

return l

inp=list(input("Enter the number :"))

print factorial\_list(inp)

#!/usr/bin/python

def factorial\_tuple(inp):

l=[]

for val1 in inp:

val=1

for v in range(1,val1+1):

val\*=v

l.append(val)

return tuple(l)

inp=list(input("Enter the number :"))

print factorial\_tuple(inp)

#!/usr/bin/python

import fact

print fact.factorial(5)

**Answer:**

gadikgop@VTA077l:~/python/day5/fourth$ ./main.py

Enter the number :13

Factorial of 13 is 6227020800

120

**Question 5:**

5. Develop Python package to include above modules of Fibonacci series, prime numbers and factorial

numbers

gadikgop@VTA077l:~/python/day5$ ls

first.py fourth \_\_init\_\_.py second third

#!/usr/bin/python

import first

from second import main

from third import main

from fourth import main

**Answer:**

gadikgop@VTA077l:~/python/day5$ ./\_\_init\_\_.py

Please pass arguments..

Enter the input from the user :10

0 1 1 2 3 5 8 13 21 34

0 1 1 2 3 5 8 13 21 34

Enter the input number :12

12 is not a prime number

20 is not a prime number

Enter the number :28

Factorial of 28 is 304888344611713860501504000000

120