**Data Structures and Algorithm**

**Practice Assignment (Python Programming)**

**Ron Jacob Varghese**

**AM.EN.U4CSE19346**

**CSE-D**

**Section A: Python Basics and Loops**

1. Write a python program to check whether a number is divisible by 7 or not.

**--->**

**x = int(input())**

if x % 7 == 0:

print("Divisible by 7")

else:

print("is not divisible by 7")

****

2. Write a python program to input marks of five subjects Physics, Chemistry, Biology,

Mathematics and Computer. Calculate percentage and grade according to following:

Percentage >= 90% : Grade A

Percentage >= 80% : Grade B

Percentage >= 70% : Grade C

Percentage >= 60% : Grade D

Percentage >= 40% : Grade E

Percentage < 40% : Grade F

NB:- Marks ranges from 0 – 100. Need to check for the invalid inputs.

**--->**

**Sub=list(map(int,input().split()))**

if min(Sub)<0 or max(Sub)>100:

print("invalid input")

else:

s=sum(Sub)/5

if s>89:

print("Grade A")

elif s>79:

print("Grade B")

elif s>69:

print("Grade C")

elif s>59:

print("Grade D")

elif s>49:

print("Grade E")

else:

print("Grade F")



3. Write a python program to find sum of all even numbers between 1 to n.

**--->**

**n=int(input());print(n//2\*(n//2+1))**



4. Write a python program to print multiplication table of a given number.

NB: If the given number is 5 then the output should be in the following format

1 x 5 = 5

2 x 5 = 10

.................

.................

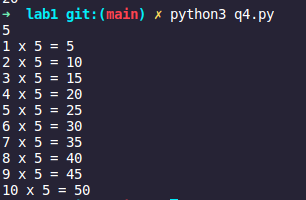
10 x 5 = 50

**--->**

**n=int(input())**

for i in range(1,11):

print(i,"x",n,"=",i\*n)



5. Write a python program to check whether a number is Armstrong number or not.

eg:- 153 is an Armstrong number. Because, 1^3 + 5^3 + 3^3 = 153

**--->**

**n=int(input());t=n**

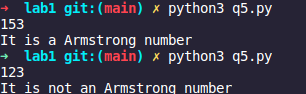
s=0

while t!=0:

s+=(t%10)\*\*3

t=t//10

print("It is a Armstrong number" if s==n else "It is not an Armstrong number")



6. Write a python program to print Fibonacci series up to n terms.

**--->**

**n=int(input())**

a=0

b=1

if n<=0:

print("invalid number")

elif n==0:

print(a)

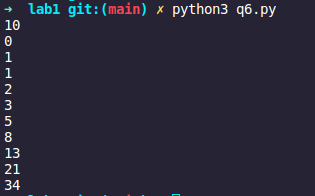
else:

for i in range(1,n+1):

print(a)

c=a+b

a=b



7. Write a python program to solve Quadratic Equation

**--->**

**a=int(input())**

b=int(input())

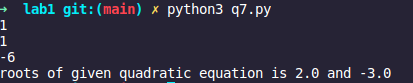
c=int(input())

d=(b\*\*2)-(4\*a\*c)

x1=(-b+(d\*\*0.5))/(2\*a)

x2=(-b-(d\*\*0.5))/(2\*a)

print("roots of given quadratic equation is",x1,"and",x2)



**Section B: Python Function**

**Write a Python program to create a function:**

8. To check whether a number is palindrome or not and returns True or False

respectively.

**--->**

**def palindrome(*n*):**

m=n

t=0

while m!=0:

l=m%10

t=t\*10+l

m=m//10

if t==n:

return True

else:

return False

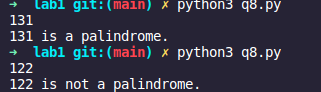
n=int(input())

if palindrome(n):

print(n,"is a palindrome.")

else:

print(n,"is not a palindrome.")



9. That accept an integer number, find if it is Disarium number or not (Hint: A number

is said to be a Disarium number when the sum of its digit raised to the power of their

respective position is equal to the number itself. Eg: 175 is a Disarium number, 1 1 +

7 2 + 5 3 = 175)

**--->**

**def disarium(*n*):**

m=n

c=0

s=0

while m!=0:

m=m//10

c+=1

m=n

while c!=0:

l=m%10

s+=l\*\*c

m=m//10

c-=1

if s==n:

return True

else:

return False

n=int(input())

if disarium(n):

print(n,"is disarium number")

else:

print(n,"is not disarium number")



10. That accepts a string, check for Palindrome and returns True or False respectively.

**--->**

**def palindrome(*n*):**

if n==n[::-1]:

return True

else:

return False

n=input()

if palindrome(n):

print(n,"is a palindrome.")

else:

print(n,"is not a palindrome.")



11. To check whether a number is Prime number or not.

**--->**

**def prime(*n*):**

for i in range(2,n):

if n%i==0:

return False

return True

n=int(input())

if prime(n):

print(n,"is prime.")

else:

print(n,"is composite.")



**Write a Python program to create a recursive function:**

12. To calculate the factorial of a number (a non-negative integer). Pass the integer as an

argument and return the factorial.

**--->**

**def fact(*n*):**

if n==0:

return 1

else:

return n\*fact(n-1)

n=int(input())

print(fact(n))



13. That accepts a number and find the harmonic sum of that number. (Hint: the

harmonic sum of n is equal to the sum of reciprocals of positive integers up to n). For

example, if n =3, HS = 1+1/2+⅓

**--->**

**def harm\_sum(*n*):**

if n==1:

return 1

else:

return (1/n)+harmonic\_sum(n-1)

n=int(input())

print(harm\_sum(n))

****

14. That accepts a number and find the sum of geometric series of that number. (Hint:

Geometric series of 4 = 1+1/2+1/4+1/8)

**--->**

**def geo\_sum(*n*):**

if n==0:

return 1

return 1/(3\*\*n)+geometric\_sum(n-1)

n=int(input())

print(geo\_sum(n))

