**TABLE OF CONTENTS**

[ACKNOWLEDGEMENT 2](#_Toc28289609)

[MINIMUM SYSTEM REQUIREMENTS 3](#_Toc28289610)

[PURPOSE OF PROJECT 4](#_Toc28289611)

[FLOW OF PROGRAM 5](#_Toc28289612)

[SOURCE CODE 6](#_Toc28289613)

[SCREENSHOTS : CODE 14](#_Toc28289614)

[SCREENSHOTS : OUTPUT 18](#_Toc28289615)

# 

# **ACKNOWLEDGEMENT**

The success and final outcome of this assignment required a lot of guidance and assistance from many people. Whatever I have done is only due to such guidance and assistance and thus would like to thank them all.

I would like to express my special thanks of gratitude to my teacher Mrs. Alpa Biju, who gave me the golden opportunity to do this wonderful project of Computer Science. She provided all support and guidance which I needed during the course of project without which it was probably not possible to complete the assignment on time. I am extremely grateful to her for providing such nice support and guidance. During the course of the project I learned about many new things.

My parents constantly boosted my morale and helped me in finalizing the project in the limited time frame. I would like to thank them. They were also a constant source of new ideas.

**Mudit Garg**

**XI – A**

**(2019 – 20)**

# **MINIMUM SYSTEM REQUIREMENTS**

**HARDWARE**

A PC with following or similar configuration:

* Intel Atom® processor or Intel® Core™ i3 processor
* 1 GB RAM or 1GB free disk space
* 1024x768 screen resolution

**SOFTWARE**

* Python 2.4 or higher, Jython, PyPy or IronPython

OPERATING SYSTEM:

* Windows:
  + Microsoft Windows 10/8/7/Vista/2003/XP (incl.64-bit)
* Mac:
  + Mac OS X 10.8 or higher
* Linux:

# 

# **PURPOSE OF PROJECT**

The purpose of this project is to apply the programming

knowledge into a real- world situation/problem and understand how

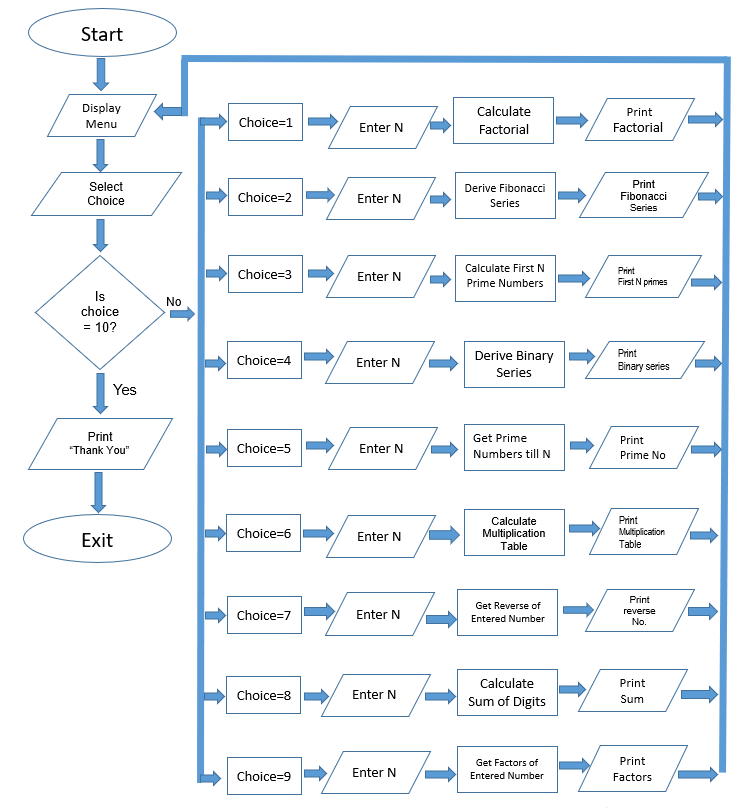
programming skills helps in developing a good software and increase logical reasoning.

Also to :

* Master the fundamentals of writing Python scripts
* Write Python functions to facilitate code reuse
* Make the code perfect by handling errors
* Work with the Python standard library
* Explore Python's object-oriented features
* Apply object oriented programming principles effectively when developing small to medium sized projects.
* Write effective procedural code to solve small to medium sized problems.
* To demonstrate a breadth of knowledge in computer science, as exemplified in the areas of systems, theory and software development.
* Demonstrate ability to conduct a research or applied Computer Science project, requiring writing and presentation skills which exemplify scholarly style in computer science.

# 

# **FLOW OF PROGRAM**



# **SOURCE CODE**

#Menu Driven Program

#1. Factorial

#2. Fibonacci Series

#3. First N Prime Numbers

#4. Binary Series

#5. Prime Numbers till N

#6. Multiplication Table

#7. Reverse entered number

#8. Sum of digits of entered number

#9. Factors of N

#10. Exit

print(" Menu Driven Program")

def menu():

print("Select which option to print : ")

print('1. Factorial')

print('2. Fibonacci Series')

print('3. First N Prime Numbers')

print('4. Binary Series')

print('5. Prime Numbers till N')

print('6. Multiplication Table')

print('7. Reverse entered number')

print('8. Sum of digits of entered number')

print('9. Factors of N')

print('10. Exit')

#factorial

def fact(n):

f=1

if n==0:

return 1

else:

while n:

f=f\*n

n-=1

return f

#Fibonacci Series

def fibo(n):

ctr=ltop = l =1

lst=[1,1]

while ctr<=(n-2):

c = ltop+l

lst.append(c)

ltop=l

l=c

ctr+=1

print(lst)

#First N prime numbers

def prime(n):

list=[]

a=[]

length=len(list)

for val in range(1000000000):

if val > 1:

for m in range(2, val):

if (val % m) == 0:

break

else:

list+=[val]

length=len(list)

if length > n-1:

print("The first", n,"prime numbers are : ")

for i in list:

print(i)

break

#Binary Series

def binary(n):

for i in range(n):

print(2\*\*i, end=' ' )

print()

#Prime Numbers till N

def primen(n):

for i in range(0,n+1):

if i>1:

for j in range(2,i):

if i%j==0 :

break

else:

print(i)

#Multiplication Table

def multi(n):

for a in range(1,11):

print(n, "X" , a, "=", a\*n)

#Reverse the entered number

def rev(n):

m= 0

while(n> 0):

p= n %10

m = (m\*10) +p

n = n //10

print("Reverse of entered number is =", m)

#Sum of digits of entered number

def digsum(n):

s=0

while n!=0:

g=n%10

s=s+g

n=int(n/10)

ans=int(s)

print("The sum of the digits of given number is : ",ans)

#Factors of N

def factor(n):

print("The Factors of the number are:")

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

if(n%i==0):

print(i)

#Main Loop

while(True):

menu()

print()

choice=int(input("Select Option Number :"))

if choice ==10:

print("Thank You !")

exit()

else:

n=int(input("Enter N :"))

if choice==1:

print(fact(n))

print()

elif choice==2:

fibo(n)

print()

elif choice==3:

prime(n)

print()

elif choice==4:

binary(n)

print()

elif choice==5:

primen(n)

print()

elif choice==6:

multi(n)

print()

elif choice==7:

rev(n)

print()

elif choice==8:

digsum(n)

print()

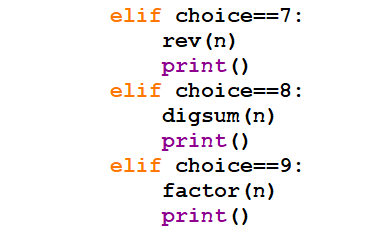
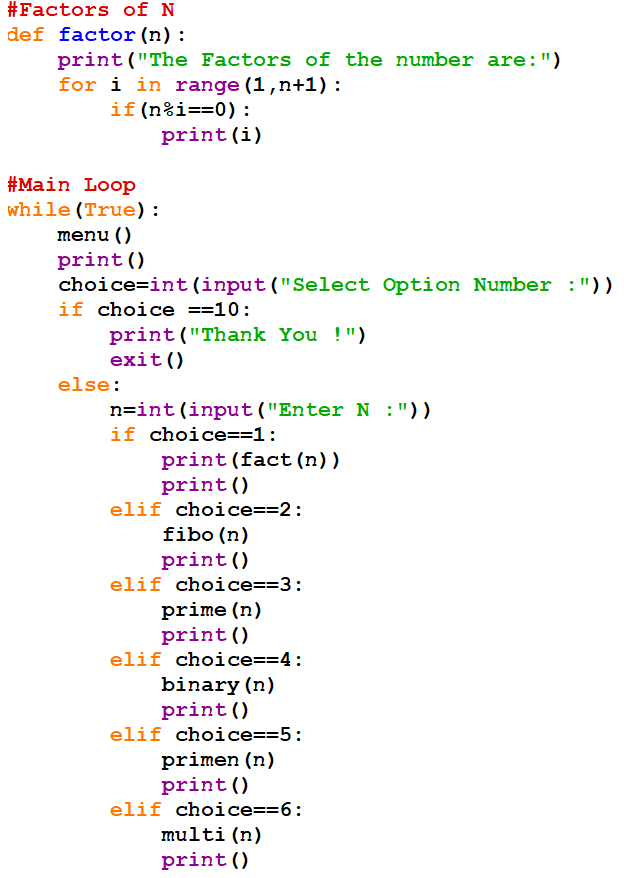
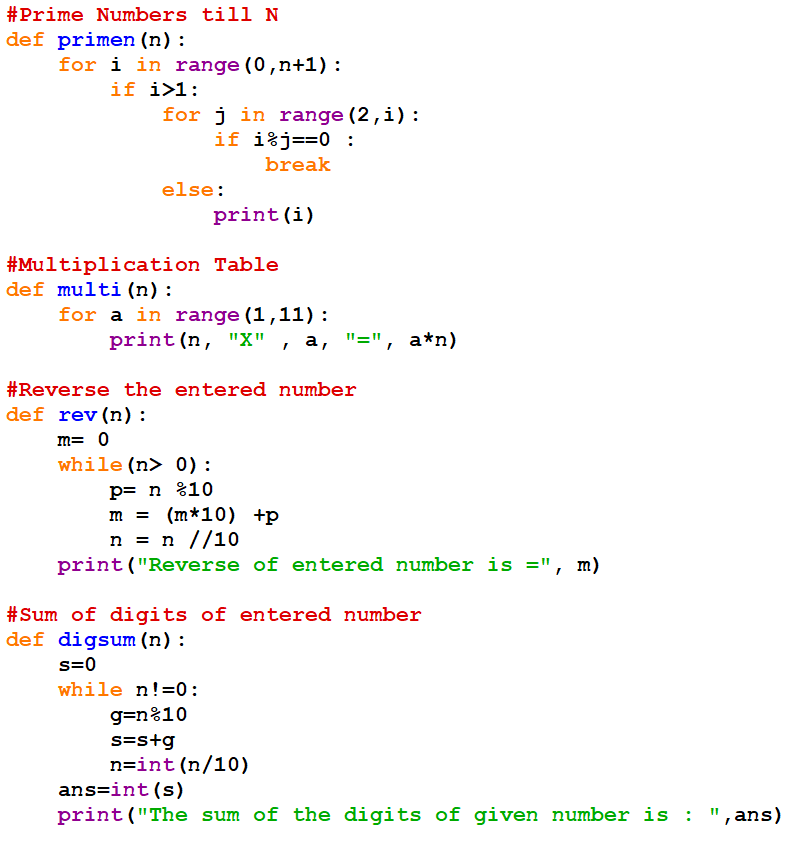
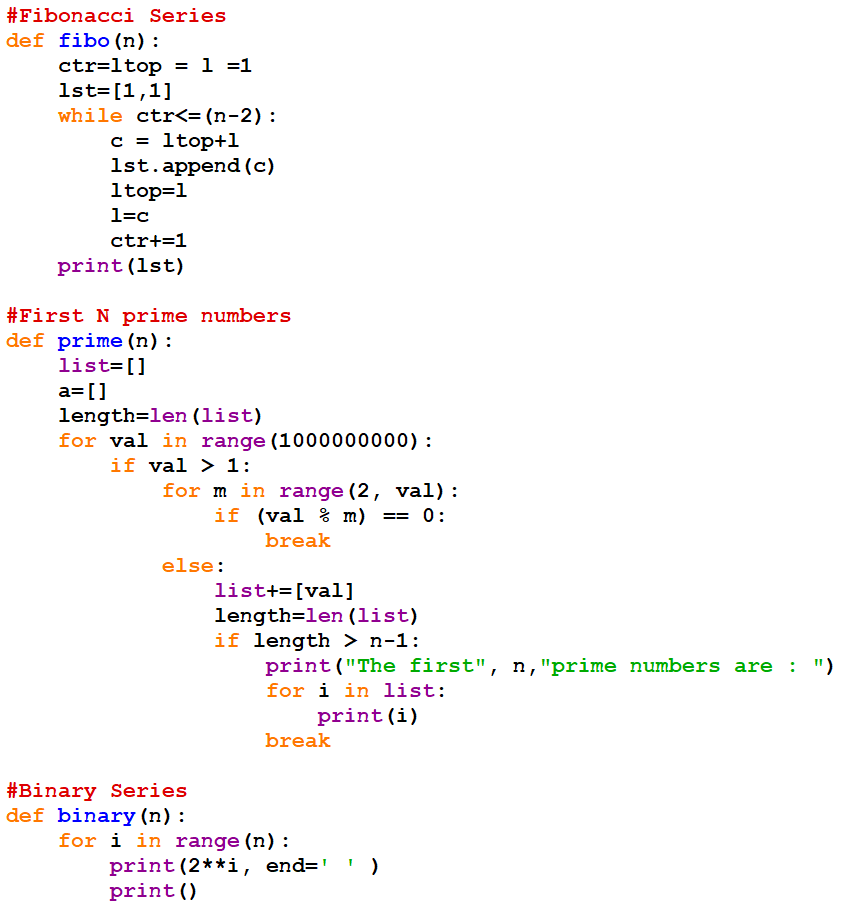
elif choice==9:

factor(n)

print()

# **SCREENSHOTS : CODE**

# 



# **SCREENSHOTS : OUTPUT**

