**Problem Statement 1:**

State Bank of India is planning to introduce Automated Teller Machine (ATM) system for its customers to provide convenient withdrawal of Cash. A software is to be designed that will control ATM dispenser for cash (in multiples of Rs.500, 100, 50, 10 and 5 only)

Write a program which will take Amount as input and gives **output** as break up of how many notes of each denomination will be dispensed.

If not able to dispense amount, it should show error as 'Invalid Amount'

**For e.g.:**

555 as input will return the output as:

500 \* 1

50 \* 1

5 \* 1

Code :

dividend = int(input())

notes = [500,100,50,10,5]

if dividend%5 ==0:

for divisor in notes:

quotient = dividend//divisor

dividend =dividend%divisor

print(“{0} \* {1} = {2}”.format(divisor , quotient , divisor\*quotient)

quotient =0

else:

print("invalid amount")

algorithm:

dividend = getting input from user

list of notes = [500,100 ,50,10 ,5 ]

checking the if the input is divisible by 5 if it is then :

iterating on list of notes:

deriving the quotient by calculation : quotient = dividend//divisor

updating the dividend by calculation : dividend =dividend%divisor

printing the divisor , quotient and the amount

again intialising back quotient to 0 for next note

suppose if the input is not divisible by 5 then:

show the user that amount entered is invalid

Write a program to find out factorial of a number. Write using both recursion and Iteration techniques.

**For e.g.:**

Input 5

Factorial is (5 \* 4 \* 3 \* 2 \* 1) so output 120

Input 6

Factorial is (6 \* 5 \* 4 \* 3 \* 2 \* 1) so output 720

RECURSION:

def fac(n):

if n ==0 or n==1:

return 1

elif n<0:

return "factorial cant be defined for negative numbers"

else:

return n\*fac(n-1)

n = int(input())

print(fac(n))

ITERATION:

def fac(n):

if n <0:

return “factorials are not defined for negative numbers”

else:

res = 1

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

res \*=i

return res

n = int(input())

print(fac(n))

Approach:

RECURSION:

Defining the factorial function:

Suppose the number is 0 or 1 :

Return user as 1

If the input is negative :

Return the user that “factorials are not defined for negative numbers”

if the input is positive number :

return the number with multiplication of its preceding num which is placed in same –function

take the input from user and call the function

Iteration:

Defining the factorial function:

if the number is -ve:

return “factorials are not defined for negative values

if the number is +ve or 0:

intialise a variable to 1

iterate a loop from 1 to the range of n+1:

multiply and assign the multiplied value to variable as res \*= i

return res

take the input from user and call the function