18CSC207J-Advance Programming Practice - Structured Programming – Lab Programs

Name :- Puneet Sharma
Reg. No. :- RA1911003010331
Class :-CSE F1

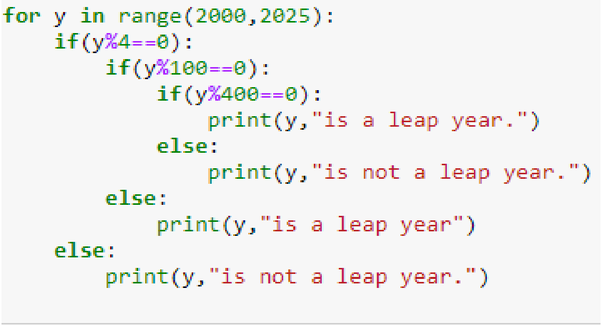
SET 8

1.Develop a python program using decision control structure find the leap years between the year limit 2000 to 2025.

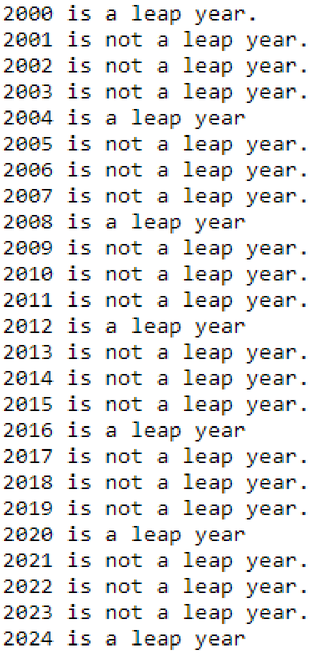
Solution.

Code:

```
for y in range(2000,2025):
    if(y%4==0):
        if(y%100==0):
            if(y%400==0):
                print(y,"is a leap year.")
            else:
                print(y,"is not a leap year.")
        else:
            print(y,"is a leap year")
    else:
        print(y,"is not a leap year.")
```



```
for y in range(2000,2025):
    if(y%4==0):
        if(y%100==0):
            if(y%400==0):
                print(y,"is a leap year.")
            else:
                print(y,"is not a leap year.")
        else:
            print(y,"is a leap year")
    else:
        print(y,"is not a leap year.")
```



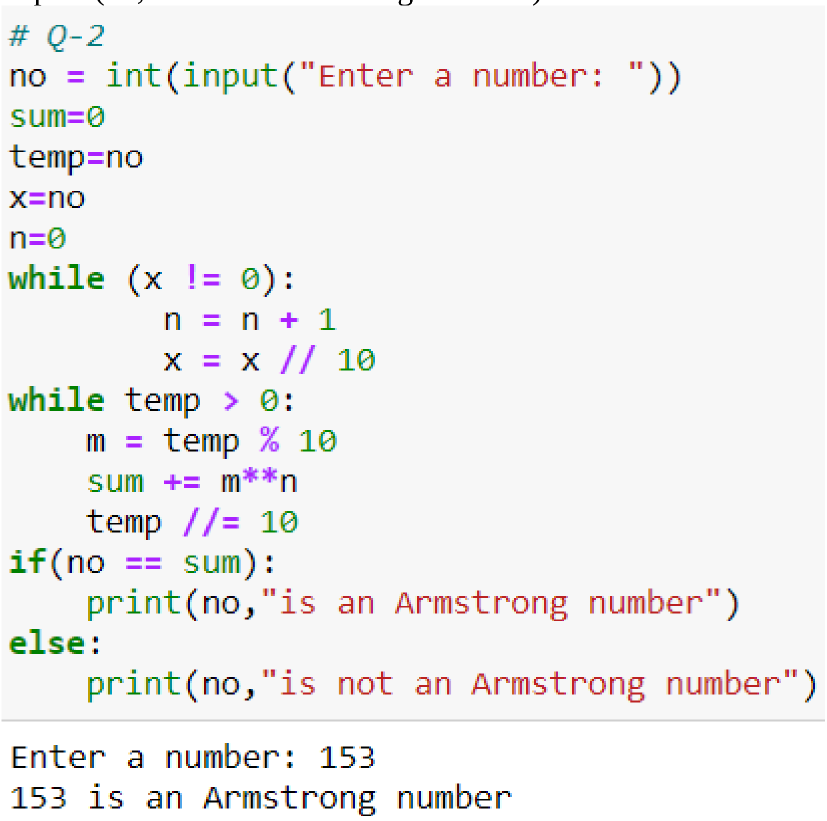
```
2000 is a leap year.
2001 is not a leap year.
2002 is not a leap year.
2003 is not a leap year.
2004 is a leap year
2005 is not a leap year.
2006 is not a leap year.
2007 is not a leap year.
2008 is a leap year
2009 is not a leap year.
2010 is not a leap year.
2011 is not a leap year.
2012 is a leap year
2013 is not a leap year.
2014 is not a leap year.
2015 is not a leap year.
2016 is a leap year
2017 is not a leap year.
2018 is not a leap year.
2019 is not a leap year.
2020 is a leap year
2021 is not a leap year.
2022 is not a leap year.
2023 is not a leap year.
2024 is a leap year
```

2.Develop a python program to check the given number is Armstrong number or not using iteration control structures.

Solution.

Code:

```
no = int(input("Enter a number: "))
sum=0
temp=no
x=no
n=0
while (x != 0):
        n = n + 1
        x = x // 10
while temp > 0:
    m = temp % 10
    sum += m**n
    temp //= 10
if(no == sum):
    print(no,"is an Armstrong number")
else:
    print(no,"is not an Armstrong number")
```



```
# Q-2
no = int(input("Enter a number: "))
sum=0
temp=no
x=no
n=0
while (x != 0):
        n = n + 1
        x = x // 10
while temp > 0:
    m = temp % 10
    sum += m**n
    temp //= 10
if(no == sum):
    print(no,"is an Armstrong number")
else:
    print(no,"is not an Armstrong number")
```

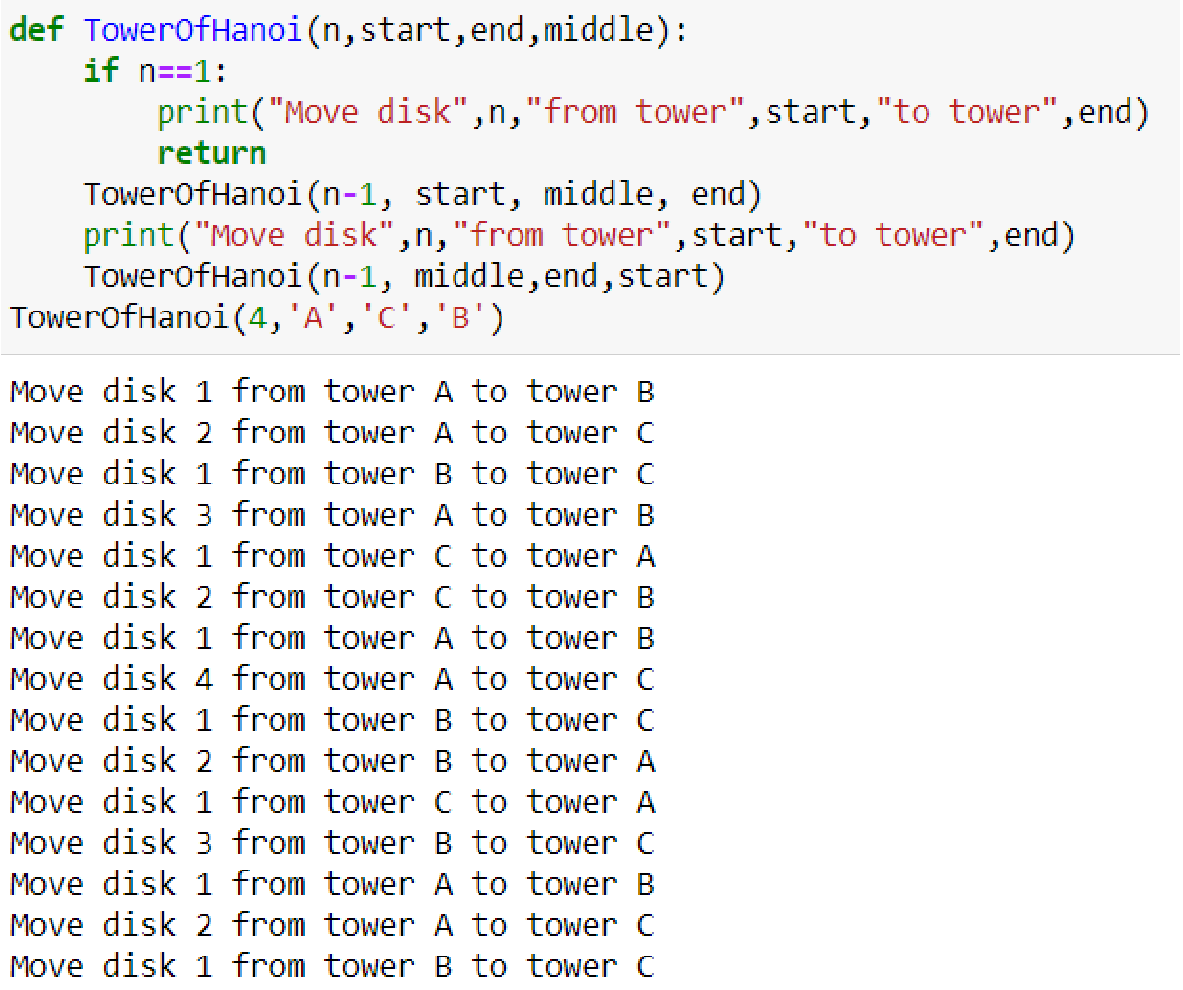
```
Enter a number: 153
153 is an Armstrong number
```

3. Solve the towers of Hanoi problem using recursive function in python language (n=3 discs)

Solution.

Code:

```
def TowerOfHanoi(n,start,end,middle):
    if n==1:
        print("Move disk",n,"from tower",start,"to tower",end)
        return
    TowerOfHanoi(n-1, start, middle, end)
    print("Move disk",n,"from tower",start,"to tower",end)
    TowerOfHanoi(n-1, middle,end,start)
TowerOfHanoi(4,'A','C','B')
```



```
def TowerOfHanoi(n,start,end,middle):
    if n==1:
        print("Move disk",n,"from tower",start,"to tower",end)
        return
    TowerOfHanoi(n-1, start, middle, end)
    print("Move disk",n,"from tower",start,"to tower",end)
    TowerOfHanoi(n-1, middle,end,start)
TowerOfHanoi(4,'A','C','B')
```

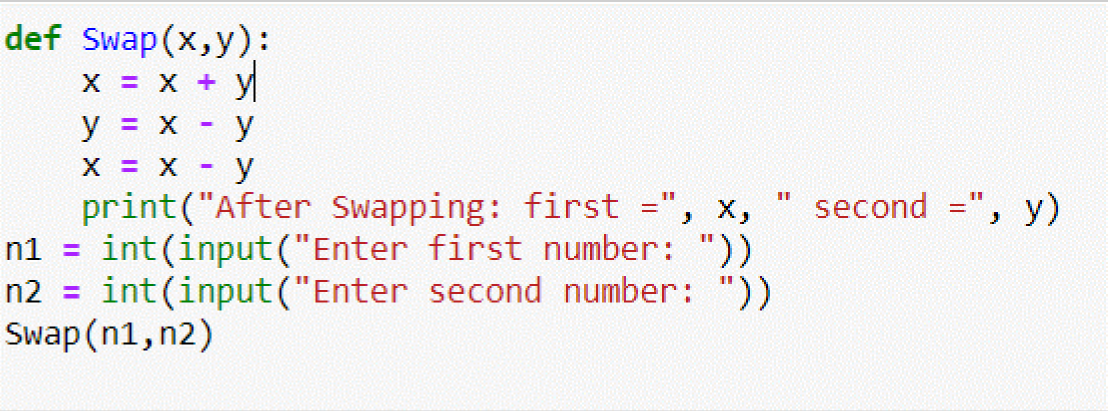
```
Move disk 1 from tower A to tower B
Move disk 2 from tower A to tower C
Move disk 1 from tower B to tower C
Move disk 3 from tower A to tower B
Move disk 1 from tower C to tower A
Move disk 2 from tower C to tower B
Move disk 1 from tower A to tower B
Move disk 4 from tower A to tower C
Move disk 1 from tower B to tower C
Move disk 2 from tower B to tower A
Move disk 1 from tower C to tower A
Move disk 3 from tower B to tower C
Move disk 1 from tower A to tower B
Move disk 2 from tower A to tower C
Move disk 1 from tower B to tower C
```

4.Develop a python code declare the function swap and perform swapping operation of two numbers without using temp variables.

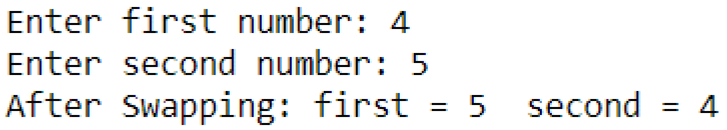
Solution.

Code:

```
def Swap(x,y):
   x = x + y
   y = x - y
   x = x - y
   print("After Swapping: first =", x, " second =", y)
n1 = int(input("Enter first number: "))
n2 = int(input("Enter second number: "))
Swap(n1,n2)
```



```
def Swap(x,y):
    x = x + y
    y = x - y
    x = x - y
    print("After Swapping: first =", x, " second =", y)
n1 = int(input("Enter first number: "))
n2 = int(input("Enter second number: "))
Swap(n1,n2)
```



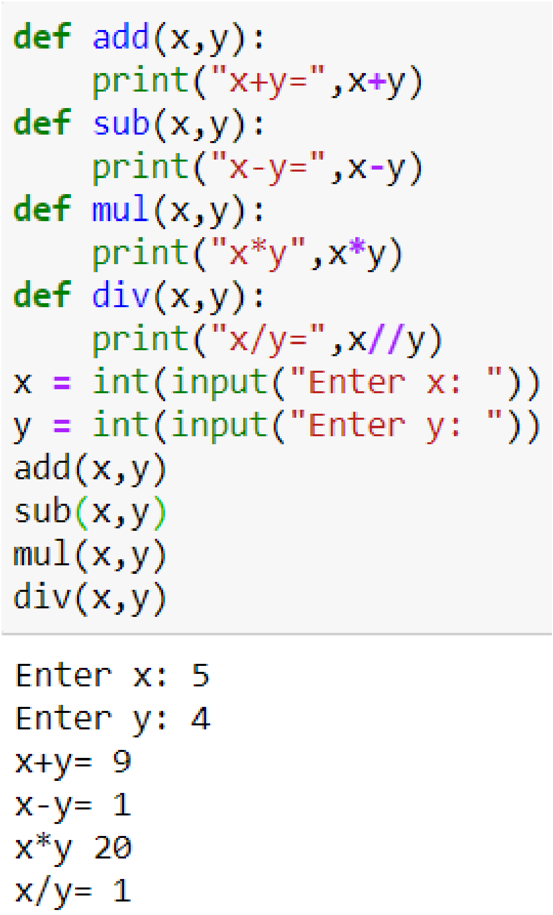
```
Enter first number: 4
Enter second number: 5
After Swapping: first = 5  second = 4
```

5. Develop a python program which should have all function prototypes to perform any four arithmetic operations.

Solution.

Code:

```
def add(x,y):
   print("x+y=",x+y)
def sub(x,y):
   print("x-y=",x-y)
def mul(x,y):
   print("x*y",x*y)
def div(x,y):
   print("x/y=",x//y)
x = int(input("Enter x: "))
y = int(input("Enter y: "))
add(x,y)
sub(x,y)
mul(x,y)
div(x,y)
```



```
def add(x,y):
    print("x+y=",x+y)
def sub(x,y):
    print("x-y=",x-y)
def mul(x,y):
    print("x*y",x*y)
def div(x,y):
    print("x/y=",x//y)
x = int(input("Enter x: "))
y = int(input("Enter y: "))
add(x,y)
sub(x,y)
mul(x,y)
div(x,y)
```

```
Enter x: 5
Enter y: 4
x+y= 9
x-y= 1
x*y 20
x/y= 1
```