Lecture 11 - Modular Arithmetic

① Linearity in modular addition

$$(a+b)\%M = [(a\%M)+(b\%M)]\%M$$

② Linearity in modular multiplication

$$(a*b)\%M = [(a\%M)*(b\%M)]\%M$$

32 Power function using Recursion

```
def power(a, b) :
    if b == 1 :
        return a
    x = power(a, b//2)
    if b % 2 == 0:
        return x*x
    else:
        return x*x*a
```

33 Leap Year or not!

```
if A % 400 == 0 and A % 100 == 0 :
    return 1

elif A % 100 != 0 and A % 4 == 0:
    return 1
else:
    return 0
```

34 Find LCM

(2,3) ⟹ 6

(9,6) ⟹ 18

```
gNum = max(A, B)
lcm = gNum
while True :
        if  A % gNum == 0 and B% gNum == 0 :
               lcm = gNum
               break
        gNum += 1
return lcm
```

35 Find GCD

(24, 32) ⟹ 8

Pseudocode :

```
lNum = min(24,32)
gcd = lNum
while gcd >=1 :
    if A%lNum == 0 and B%lNum==0 :
        gcd = lNum
        break
    lNum -= 1
return gcd
```