```python

################################
#
#          数学系
#
################################


# Combination
class Combination:
    def __init__(self,N):
        self.fac = [1]*(N+1)
        for i in range(1,N+1):
            self.fac[i] = (self.fac[i-1]*i)%mod
        self.invmod = [1]*(N+1)
        self.invmod[N] = pow(self.fac[N],mod-2,mod)
        for i in range(N,0,-1):
            self.invmod[i-1] = (self.invmod[i]*i)%mod

    def calc(self,n,k):#nCk
        return self.fac[n]*self.invmod[k]%mod *self.invmod[n-k] %mod


#最大公約数
def gcd(a,b):
    while b:
        a,b = b, a%b
    return a

#最小公倍数
def lcm(a,b):
    return a*b // gcd(a,b)


# なんか早い素数判定
def is_prime(x):
    if x < 2: return False # 2未満に素数はない
    if x == 2 or x == 3 or x == 5: return True # 2,3,5は素数
    if x % 2 == 0 or x % 3 == 0 or x % 5 == 0: return False # 2,3,5の倍数は合成数

    # 疑似素数で割る
    prime = 7
    step = 4
    while prime <= math.sqrt(x):
        if x % prime == 0: return False
        prime += step
        step = 6 - step

    return True

```