**西安电子科技大学网信院**

**信息安全基础与密码学**

**综合实验**

**实 验 报 告（二）**

**中国剩余定理**

**班级：2118021**

**姓名：盖乐**

**学号：21009200991**

**日期：2024.9.28**

一、实验目的

1. 实验环境：
   1. Windows11
   2. Python3.12
2. 实现目标：

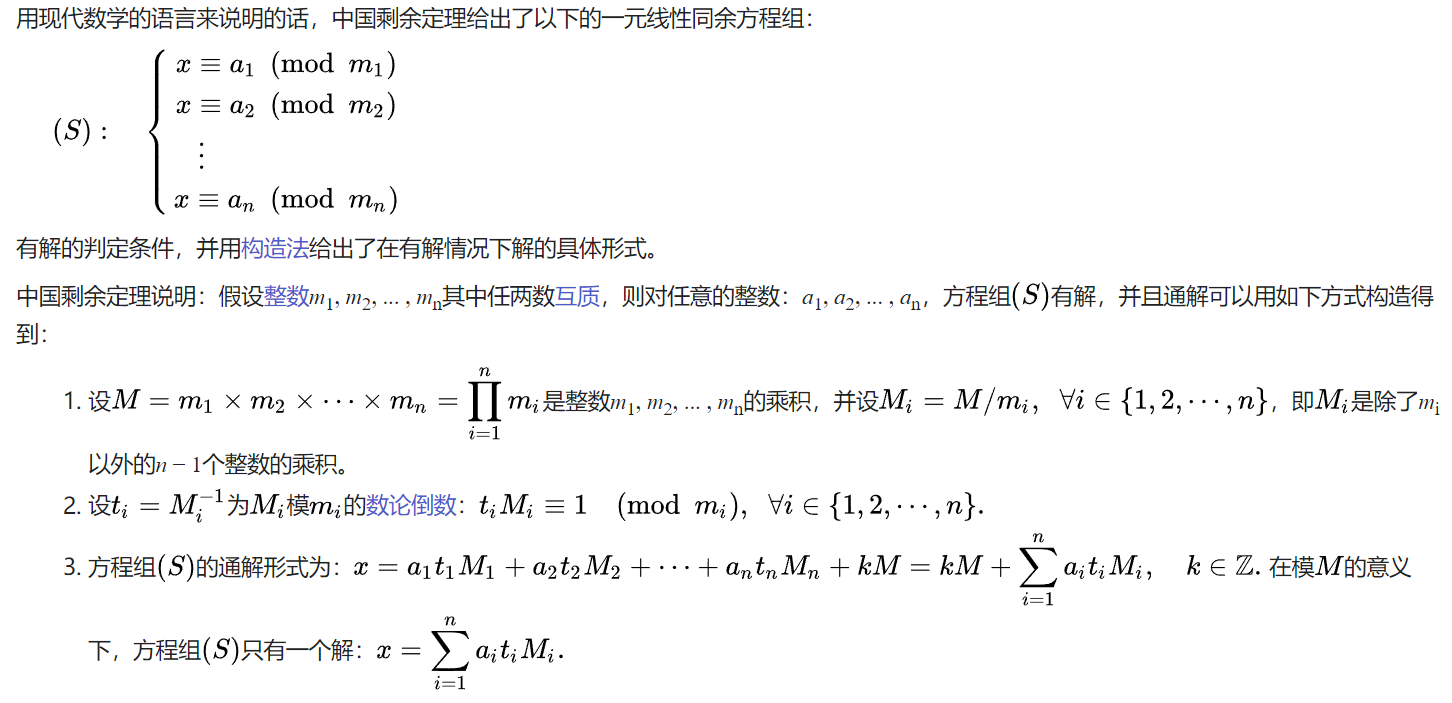
编程实现利用中国剩余定理求解一次同余方程组的解。

二、方案设计

1. 背景：

中国剩余定理，又称孙子定理或中国余数定理，是数论中的一个关于一元线性同余方程组的定理，说明了一元线性同余方程组有解的准则以及求解方法。

1. 原理



1. 算法步骤
   1. 判断正整数m1,m2,……,mk是否两两互素；是则继续，否则跳出，输出“不能直接利用中国剩余定理”；
   2. 计算m=m1m2……mk, Mj = m/mj;
   3. 计算Mj-1(mod mj)
   4. 计算xj MjMj-1aj(mod m)
   5. 计算x xj(mod m)

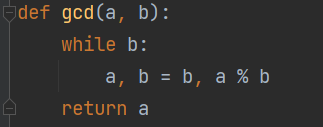
三、方案实现

1. 算法流程图

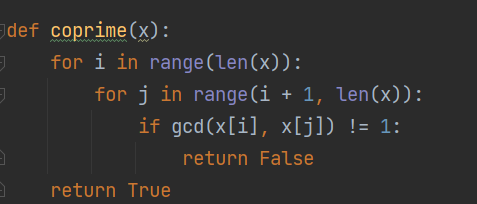
图示

描述已自动生成

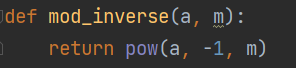
1. 主要函数的介绍
   1. 求最大公约数：



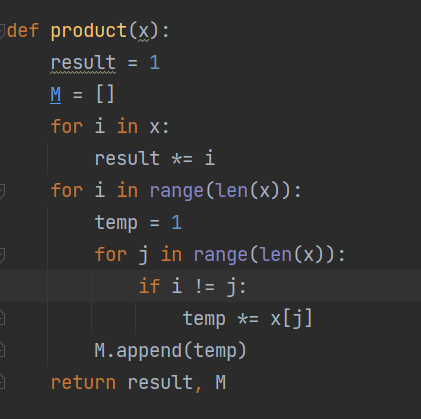
* 1. 判断是否两两否互质



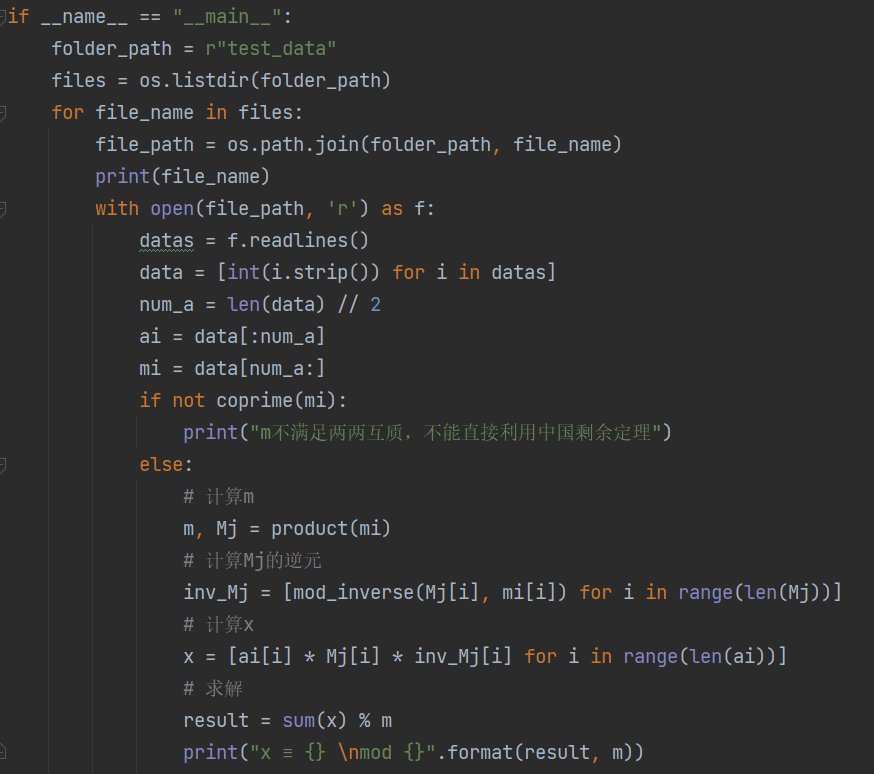
* 1. 求模m的a的逆元



* 1. 计算乘积，包含m和Mj



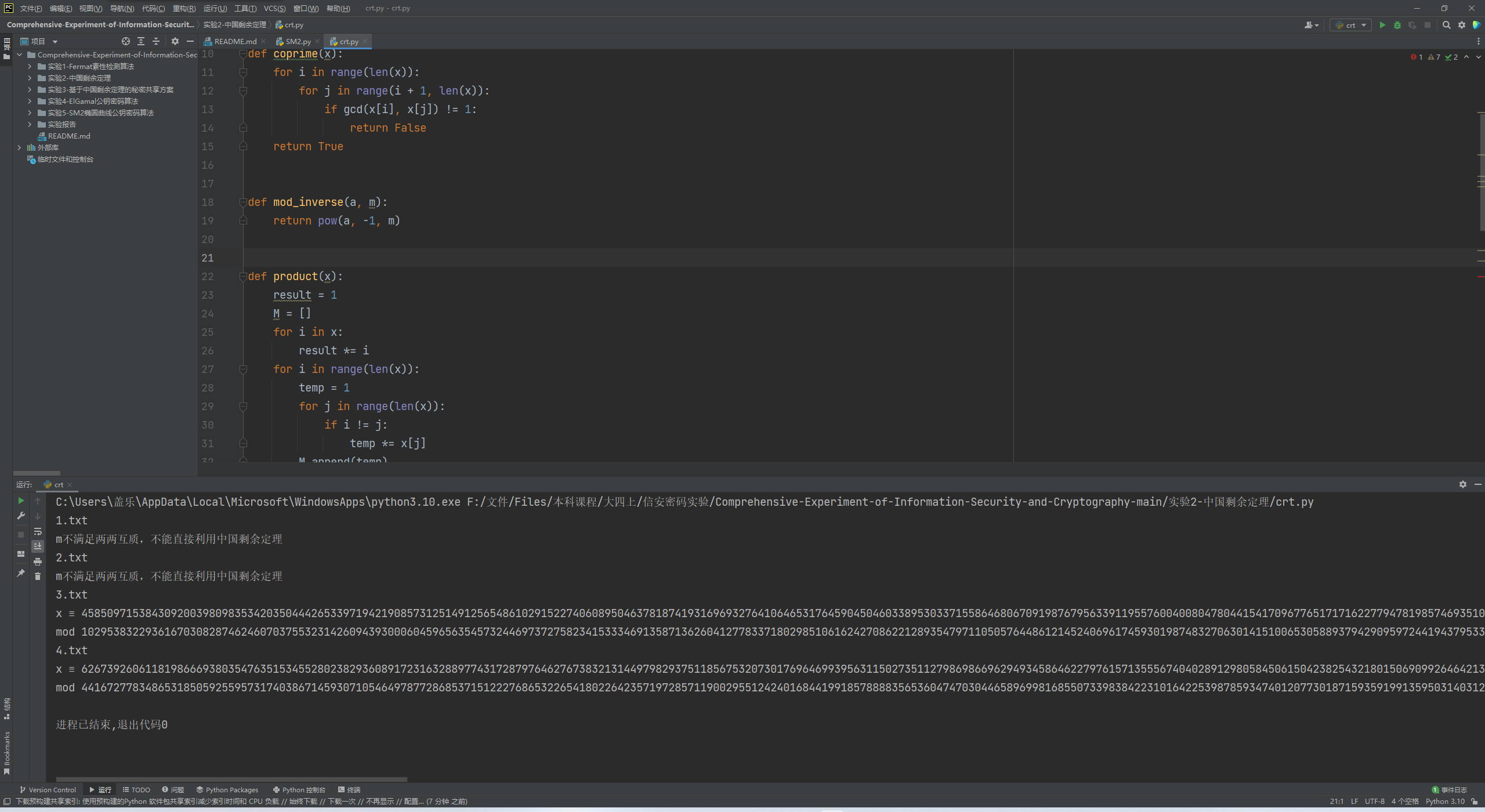
* 1. 主函数，进行文件的读取和求解



1. 算法实现的主要代码

import os  
  
  
def gcd(a, b):  
 while b:  
 a, b = b, a % b  
 return a  
  
  
def coprime(x):  
 for i in range(len(x)):  
 for j in range(i + 1, len(x)):  
 if gcd(x[i], x[j]) != 1:  
 return False  
 return True  
  
  
def mod\_inverse(a, m):  
 return pow(a, -1, m)  
  
  
def product(x):  
 result = 1  
 M = []  
 for i in x:  
 result \*= i  
 for i in range(len(x)):  
 temp = 1  
 for j in range(len(x)):  
 if i != j:  
 temp \*= x[j]  
 M.append(temp)  
 return result, M  
  
  
def select\_random\_elements(input\_list, m):  
 # 确保 m 不大于列表的长度，以避免错误  
 m = min(m, len(input\_list))  
  
 # 从输入列表中随机选择 m 个元素  
 selected\_elements = random.sample(input\_list, m)  
  
 return selected\_elements  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 folder\_path = r"test\_data"  
 files = os.listdir(folder\_path)  
 for file\_name in files:  
 file\_path = os.path.join(folder\_path, file\_name)  
 print(file\_name)  
 with open(file\_path, 'r') as f:  
 datas = f.readlines()  
 data = [int(i.strip()) for i in datas]  
 num\_a = len(data) // 2  
 ai = data[:num\_a]  
 mi = data[num\_a:]  
 if not coprime(mi):  
 print("m不满足两两互质，不能直接利用中国剩余定理")  
 else:  
 # 计算m  
 m, Mj = product(mi)  
 # 计算Mj的逆元  
 inv\_Mj = [mod\_inverse(Mj[i], mi[i]) for i in range(len(Mj))]  
 # 计算x  
 x = [ai[i] \* Mj[i] \* inv\_Mj[i] for i in range(len(ai))]  
 # 求解  
 result = sum(x) % m  
 print("x ≡ {} \nmod {}".format(result, m))

四、数据分析



1.txt

m不满足两两互质，不能直接利用中国剩余定理

2.txt

m不满足两两互质，不能直接利用中国剩余定理

3.txt

x ≡ 45850971538430920039809835342035044426533971942190857312514912565486102915227406089504637818741931696932764106465317645904504603389530337155864680670919876795633911955760040080478044154170967765171716227794781985746935101945954494367888075899401250020053828724797519906076403988854722257909423631915311803168435017404946975580870213059707263130497544483848012050936544965684165927589759568960270987203441010266895562059551047160497042549580208730503986196805685742898949454146252047342004263899297290352022925068888842847365514463830992010079124868369216920001186960543284874385035998480029862150276601705976521890712672606441884107284501236336640253336010280687676722387082266736122849510381088976153557510307378833176629203010712055604988133944905160586176524594316889459005908884571099464152689527352656388293853679883196076846081112428967757311121854885585398974679164977769195152760696616838930

mod 102953832293616703082874624607037553231426094393000604596563545732446973727582341533346913587136260412778337180298510616242708622128935479711050576448612145240696174593019874832706301415100653058893794290959724419437953367492390340966268444747921797002666302065419051284327316436596511610022028215683792605669607405846246497742157867114795477989041761794634036907583854616979469929197455574810495931418179547756207701750504953681908376067692967274515542008467677105644775319423602660757912702485599517785360716305908479237359822322474824561080012142075430498482341564681447500226439508529311218175599586292185893671487692689314204073453837111333506885062212732994672392190791606071313185669874404097203502329489287292532064710265497653276514454976780972201788622137736651988293004202789504266484385346148359440631518283752042877070987341306359791629867004079011333875796459161265020098175460251658121

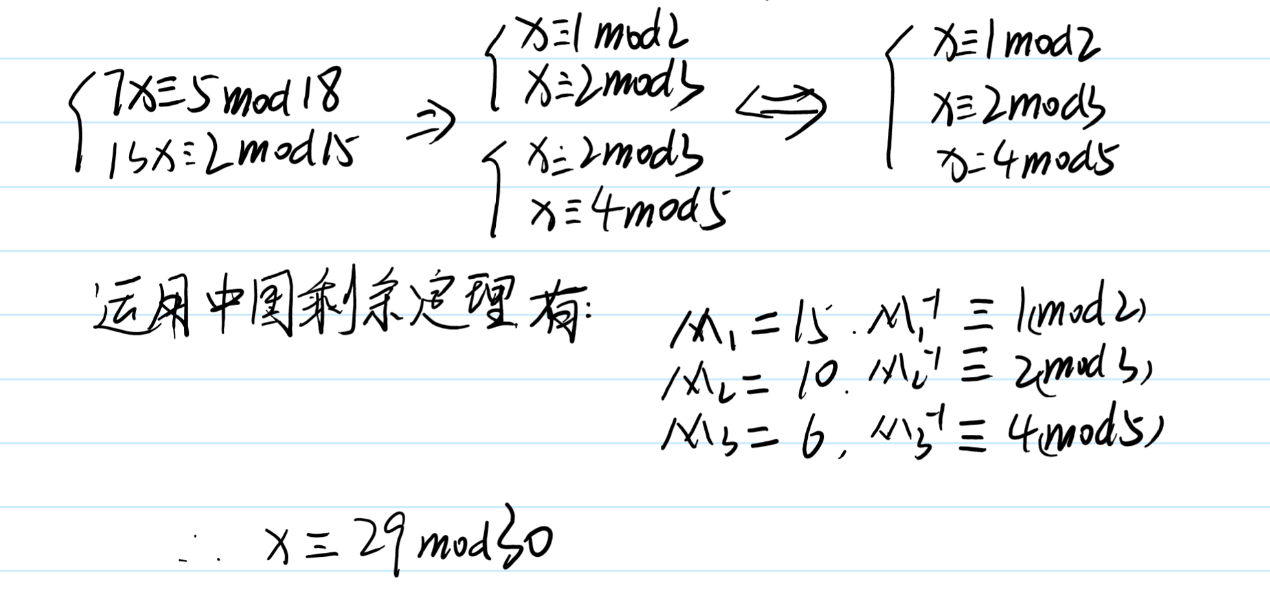
4.txt

x ≡ 6267392606118198666938035476351534552802382936089172316328897743172879764627673832131449798293751185675320730176964699395631150273511279869866962949345864622797615713555674040289129805845061504238254321801506909926464213324484282320421714571228986528089352506899736933010662849746895547877896800919085622299496414516521109484070814786174506514498019165273501727938625402571547656353590057030677574223761865456469779358997466768453711269097718055367640778587218315980747316831735149013184044851096369030486987130491601121502670977591761060152995444043920595601104637105776654836634982589836094359514364557075424100659668050103473452225458227798148841038514448303792904955511004133401282914950720654749341284341008310482855102942461387434475549098702381300184287945365098142095772121750282771162438266376091878150477584642296687481954338735808894790421992792101115025793836754451597837211648884480656

mod 44167277834865318505925595731740386714593071054649787728685371512227686532265418022642357197285711900295512424016844199185788883565360474703044658969981685507339838422310164225398785934740120773018715935919913595031403123730195548868895596981780364410766438153687334085748954703628390494088766685638724111099153395397990775139019064220117946529126064087603370677680157881819738461102417259553876319102288706171985087178697972986242540571926766725811581285490145946480434821092468831574238874260238224160478687257261737726163348031263191488802014458498928371530579204439498493860479203458297644637699484139562127849767251451367401649996295995201735622769926331841381668172373020987044031367797353154923281862023390934881926814699658430510263517934218266652711222603092764613010527837639734437073454613947384979607624431521137936390961606575979334870290494119935029783148127265051014571924493060770273

五、思考与总结

1. 求一次同余方程组的解，若正整数𝒎𝟏，𝒎𝟐，…，𝒎𝒌不是两两互素，是否能直接用中国剩余定理求解？例如方程组，需要如何求解？

不能。需要先将其转换为中国剩余定理所需要的形式，即x前面系数为一，正整数𝒎𝟏，𝒎𝟐，…，𝒎𝒌两两互素，具体过程如下：

1. 实验过程中还遇到了什么问题，如何解决的？通过该实验有何收获？

文本

描述已自动生成

Python的除法操作溢出了，转换为乘法，即在计算m的过程中算出Mj来，而不是通过m/mj得到。

