1. $q = \lfloor a/b \rfloor$

$\Rightarrow a = bq + r.\ q \in \mathbb{Z},\ r < b.$

$a \geq b > 0:\ q > 0.$

$l(a) - l(b) - 1$

$= \log_2 |a| - \log_2 |b| - 1$

$= \log_2 \frac{a}{b} - \log_2 2$

$= \log_2 \frac{a}{2b}$

$l(a) - l(b) + 1$

$= \log_2 \frac{a}{b} + \log_2 2$

$= \log \frac{2a}{b}$

$\log_2 \frac{a}{2b} \leq \log_2 \lfloor \frac{a}{b} \rfloor \leq \log_2 \frac{2a}{b}$

$\Rightarrow l(a) - l(b) - 1 \leq l(q) \leq l(a) - l(b) + 1$

Discrete Mathematics: Homework 3

(Deadline: 8:00am, March 11, 2022)

1. (15 points) Let $a, b \in \mathbb{Z}$ with $a \geq b > 0$, and let $q = \lfloor a/b \rfloor$. Show that $\ell(a) - \ell(b) - 1 \leq \ell(q) \leq \ell(a) - \ell(b) + 1$, where $\ell(x)$ is the length of the binary representation of an integer x.

2. (25 points) Implement EEA (Extended Euclidean Algorithm). Run your program on the integers a, b to find two integers s, t such that $\gcd(a, b) = as + bt$, where

$r_0 = (a, b)\begin{pmatrix}1\\0\end{pmatrix}$

$r_1 = (a, b)\begin{pmatrix}0\\1\end{pmatrix}.$

$r_2 = r_0 - q_1 r_1 \cdots$

$\begin{cases} q = \lfloor \frac{r_0}{r_1} \rfloor \\ r = r_0 \bmod r_1 \end{cases}$

until $r = 0.$

$= (a, b) \cdot \begin{pmatrix}s_0\\t_0\end{pmatrix}$

$q_n = \lfloor \frac{r_{n-1}}{r_n} \rfloor$

$r_{n+1} = r_{n-1} \bmod r_n.$

$= (a \cdot b) \cdot \begin{pmatrix}s_{n-1}\\t_{n-1}\end{pmatrix} - q_n \begin{pmatrix}s_n\\t_n\end{pmatrix}$

a=1668022384651447825852593457833359953985771134637730126520497011165389239767604
3794016150507259410995658188057040712085903607220122413595420007489488405731334280
0619883956087790107134112871312954281798133333599770341730923355794098107424397318
7888918744525312690484251399035467998130997222733657507954841157445405713326194
8502170654953266704862335547650976687291747849350782598464591428327947848142796066
9819408485961217770484110570494262217083738133966614498824146432614678060378894408
4253338496818062027178501005792458736618594429715531979857057707077347412997210
7871623872384643401132513116574551025071336188925411;

b=1785577029987051936724205968139042441809618215534204113748879687967110747874357
2864002383145021454681629377265833889126584206834902794697518171431222912791170447
5670408710944900520674073067986613374905921991707179698185015217674585778181924994
5724578050391808744973941056991119405066589753280795931975086826490329981924275
1930003066441776015464336357481344549028678389909625259705769654505066857444104947
1926476671086057147242990292233548660429548075415889373254112490970960683335559765
9869894760833106357228220147202929905178751532801162862508796644970253415643626
6476618723897816432054896528012909122280046552133534.

(**Remark:** Submit your program. The programming can be done with Python, C or C++.)

3. (25 points) Implement the Square-and-Multiply algorithm. Run your program to compute $a^e \bmod n$, where

$a \in \{0, 1, \ldots n-1\}$

$k = 7$

$e = (e_{k-1} \cdots e_0)_2.$

a=2643001830466169822724488955091646831748945577895632859292198346969979230916366
5193972706206594036869415691968221117606771494540098970766552365207210568611105852
6406300404125432978424624345267880818520745429461144042790537899763978754350060940
2906509369567325556260705033614842470769801208547000223369822886234673876359912
0210887024055251199687451392437357330469313875769415203278005429487989371958004062
1353849886761870927539333464667851350696825922397697396168849356122454249747366663
2914249190933019899352103274892031942746819319736378985973840294119088347050293
4385251934875320122360082927644910373611459923294476;

e=1440940598216013205825255507197539386591946416564947793531697088969116191795293

4. (1) $\gcd(17,23)=1$

$17x \equiv 11 \pmod{23}$

$17x \equiv (11+23) \pmod{23}$

$17x \equiv 34 \pmod{23}$

$\therefore\ x \equiv 2 \pmod{23}$

(2) $55x \equiv 35 \pmod{75}$

$\Rightarrow 11x \equiv 7 \pmod{15}$ $(7+15=22)$

$11x \equiv 22 \pmod{15}$

$\therefore\ x \equiv 2 \pmod{15}$

$x \equiv 2 \pmod{15}$

833840242062616984986924019981734081878585766104090252117790252286565595931595502
729633365857562567917164964823748671510787403884808014676043180816004775826788681
656315946088127545330496208859875078994760276323153649880368941500824854230698399
058587273230306744276048593948353049920675092662363221833779360830549535347779793
705521310372254828708923967502999845523783712266543178848696339228233321889730553
658193585853483170561690950661460813726532858449649020997668351053943818441861942
1230489065033982087166936851293061923363455338233631;

n=6454313945264858380477703362750179103894280648074641679882475733796493188829653
940877532537389629620183301943333659170185060419295800903851882920771678506908477
673738912708560686143515108791497878950835462108643709804848978316528866309066793
095973807053237106244098640248269616792697037137207037826580927776615573507736400
136484378662896553468052081722791343589348903943822231956595028500968946488659653
138113699743321196084282674797868993406360468278824654992876075514546905176286602
291631523433342533346644133635496466500102652351900303276417412474450899876006942
5321286184310908109489080474275209430911312055696378.

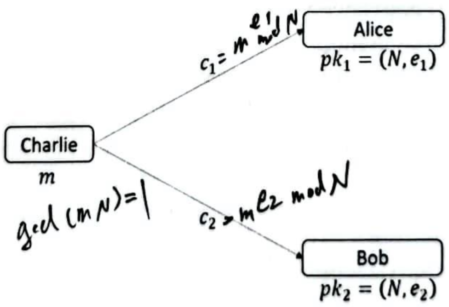
(**Remark**: Submit your program. The programming can be done with Python, C or C++.)

4. (20 points) Solve the following linear congruence equations:

   (1) $17x \equiv 11(\mathrm{mod} 23)$;

   (2) $55x \equiv 35(\mathrm{mod} 75)$.

5. (15 points) See the following figure. Alice and Bob trust each other very much. They set their RSA public keys as $pk_1 = (N, e_1)$ and $pk_2 = (N, e_2)$, respectively. Charlie wants to send a private message m to Alice and Bob, where $0 \le m < N$ is an integer and $\gcd(m, N) = 1$. To this end, Charlie encrypts m as $c_1 = m^{e_1} \bmod N$ and $c_2 = m^{e_2} \bmod N$; and then sends c_1 to Alice and sends c_2 to Bob.



密文

Suppose that $\gcd(e_1, e_2) = 1$ and Eve sees all public keys and ciphertexts. Determine if Eve can learn the value of m.

5. $\begin{cases} c_1 = m^{e_1} \bmod N \\ c_2 = m^{e_2} \bmod N \end{cases}$

$\gcd(e_1, e_2) = 1$

by EEA.

can find $e_1 s + e_2 t = 1$.

$0 \le m < N$, $\gcd(m, N) = 1$.

$\Rightarrow m \bmod N = m$.

$\Rightarrow m = m \bmod N = m^1 \bmod N = m^{e_1 s + e_2 t} \bmod N$ 2

$= ((m^{e_1} \bmod N)^s \cdot (m^{e_2} \bmod N)^t) \bmod N$

$= (c_1^s \cdot c_2^t) \bmod N$

$\Rightarrow m = (c_1^s \cdot c_2^t) \bmod N$

c_1, c_2, s, t known, Eve can.

```python
def EEA():
    a = 166802238465144782585259345783335995398577113463
    b = 178557702998705193672420596813904244180961821553
    ri = []
    qi = [1]
    si = [1,0]
    ti = [0,1]
    ri.append(a)
    ri.append(b)
    while True:
        qi.append(ri[-2]//ri[-1])
        ri.append(ri[-2]%ri[-1])
        if ri[-1] == 0: break
        si.append(si[-2]-qi[-1]*si[-1])
        ti.append(ti[-2]-qi[-1]*ti[-1])
    print("s:",si[-1])
    print("t:",ti[-1])
```

```
>>> EEA()
s: 5269346517404759757917406408306120657576139865693511443081124356069506630695623770063846774138034451326098362590654519415480012
6707869242528199250303471171536207597896008405650134889458156325490296036336342644796958477425288398387518178265890700656305714837
368523496597321973212197144244237647291270529201589
t: -492243560255702057526403691131975897841924953624400842010877571934372127411189600245929166789508023429245341157895432426179365
1077186663625890948400350842512853060168116459859792483937224361285850400246381718448690438802997126844191121984884459076214105581
3365169533361189741247565502362579257453658280613873
```

```python
def SqAndMuti():
    a = 2643001830466169822724488955091646831748945577895632859292198
    e = 1440940598216013205825255507197539386591946416564947793531697
    n = 6454313945264858380477703362750179103894280648074641679882475
    x = a
    k = len(str(bin(e)))-2
    ae = x**(e%2)
    e = e//2
    for i in range(k-1):
        ae *= ((x**2)%n) ** (e%2)
        x = (x**2) % n
        e = e//2
    print(ae%n)

SqAndMuti()
```

```
>>> SqAndMuti()
1948938994538604160707108181724192091954263523362311673846915505520625915922643693886546508713351109692750915684157878314121214348
9199923529097996539792654733505278706812520830942209991900318336435802408907249020763770922682237250909513951994814724102553142432
6059166502091869304438173719943244423806182390608997702096989971134105963997915957273941960090533678167318836865046871071816483210
9499409767199530541904080512081403155559058709882347747147418230358814131381147208291328747857991048977465984265721979324595417184
7503170017151440737380478840189460378458005476484742953848813170374548455806977675820760128018344
```