**实验一 （密码学相关数学基础）**

【实验目的及要求】

1、通过本次实验，熟悉编程环境，为后续实验做好铺垫。

2、回顾数论以及有限域上的的基本算法，加深对其理解，为本学期密码学课程及实验课打好基础。

3 、通过本次实验， 了解上元素的性质及其四则运算。

4 、掌握上的不可约多项式的判定和生成算法。

5、【必做】部分第5题需要在报告中给出算法流程图和伪代码。

# 【实验内容】

在完成必做项的基础上，从选做项任选若干项完成。

注： 1） 有限域题目中给定的不可约多项式为**0b100011011**；

2） 上的元素运算中，除模幂运算中幂指数为十进制外，其他均为十六进制表示。

【必做】

1、 欧几里得算法

实现 GCD 算法和扩展 GCD 算法，对于给定输入 a 与 b，前者只需给出最大公约数(a,b)， 后者需给出满足等式 ax+by=(a,b)的三个值：x,y,(a,b)；比较二者时间空间效率的不同。

测试点：1）7,5；

2）31,-13；

3）24,36；

4）

2461502723515673086658704256944912426065172925575，

1720876577542770214811199308823476528929542231719；

5）

13709616469144948883512229123502305176385931810284088906755090238431898972708904439178898468021710798401875986657125211084472621499595371254346390738382042，

19235039994987625167590963480899777255933775238312044097122773255647530276806317636026727679800825370459321617724871515442147432420951257037823141069640181；

6）

96557807278640299121519463045206377934978887298086994211590515571717325955785923783159432436307870512742354877476790046891802153053719263845602618422474671707896136814707875793300040916757228826108499490311295942553478010913043680523612655400526255290702983490382191419067057726624348815391509161304477322782，

146116799305702219220540123503890666704710410600856387071776221592477256752759997798169931809156426471243799795374072510423645363680537337813774268658907130969994146783451692837222772144941434909050652825715582967684984814095461041109999161468223272534833391335036612863782740784573110824091866969655931097032；

2、 快速幂取模

实现常规幂取模算法和快速幂模算法，实现常规幂取模算法和快速幂模算法，对于给定输入 a,b,c,要求输出为 a % c；比较二者时间空间效率的不同。

测试点：1）7,16,3;

2）5,1003,31

3）

1494462659429290047815067355171411187560751791530，

65537，

2268838711304724304304396119509416774597723292474；

4）

22490812876539885046336053040043361022772062226905764414319531416752624982967181455912526153033030222985778230314070837549143068021815197910334221004333099，

65537，

26381036806254391211255825330031625908895486635496820170811397576118892705526151526139312916798859030242219181178517837920904022720459931859633170905729517；

5）

237218075278892229535140238768762235405145645557640724744207466370544846457682663369976322798944392433104280595584635896821245048737637289361896703300454795175488861724813324867455119120284612785871304351940501930714775024417724051440337510897547661217466354700893011496892348407228806138461120064957907686566，

65537，

349972806688784936669965759420500287481274799328355633592840001661382340587247200055746522814275902430370330954725697648747610084477917676220179203273361291098368287612837135979510900982047154261023406927515096043384562410643544643505195484211397819374480917731785250826080723518532061522456937734714740424476；

6）

448491664748214835887077572737743989471818983924746533195711112723370968680112145505675868905886297697651811535959153343019297445815718781370807622258565317729681110144781420923700723193680834808549790079348059612669341617763791748262779560287414722951999863579554855226385841903174595011558143906566236113746355880815410935162653615576832860612181499713446185302492149321184607038850277，

266848195381815818463717950266554236453862598799637312683425703733349860929573593996414020829279843574050749733141808826377733991992433000295290381444934817157435190251178268111713086611665708753888640498769964215272509919856595016440707694201919276657450513307401642404169770745634306482286288562684063317756864776774548994812022552274677735053131675109617882635817658233698378909301711242970820585209428037532351125028227556492657705501994644156977193457255573644987541990311834672767028439520378145222935082885623390192713665176848108677291865357438200，

26449610480694582663087914798262349275307583705769208320615999887968533547578043032399193471649467130397212338172281740898344480836053483980415141663259446884375373371451231004101622624801199838411707260636384692208754088842619258012627585405663559933995516981379631336446313020148817646717985549051301115238527677472914852742788256890259402224899453419484216558327523122341749054612967901747155276100157913910547784136439888489952724508554613632641420487039242881743232756168292709989984925436911267322885917953348064673021283822937158706678666372103627074163021260578078017304088904154859161289037070912220207946945；

3、 中国剩余定理

实现中国剩余定理，对于式 x%ai=bi，对于给定的 a1，a2，a3 和 b1，b2，b3，求 x；

测试点：1）23,28,33; 0,0,0;

2）23,28,33; 5,20,34;

3）23,28,33; 283,102,23;

4)

489808178709479466279507878773770708214878979673，

896234965496726578561614071442814700467907036641，

1213827005758305602466882992172310409456053868843；

802310684485241212312289432691586430708135062249，

961714109955647014172499578071923389425123540027，

1381194006087304024683552712488022595194097928701；

5）

8157969540288411637818433039558323184074779086100165504668538221920170369774913261335059602331627321130656458962980224196880533337839226059601303464776145，

9699616044315021194953572561076502992783130623216574220426043600142343504101508838526221359049417564415801914072315788919275792502477693022853881785198116，

7832693802256371667866514213119452199821916193668106904135812283217637737600922381702016472708855675649121271702977408217814917908566132517503707494037556；

13392316081651420877308875276166772808601812122052371442339078877740399569281672683820206196320955005869072002883847646526584107260355414977120453263391947，

9734466939658282823343760206593283968765904848250021580218634383869090913086348857668999272399075016287736914000854272239315769632719896968098820774563511，

9460200357790728398862913232664036038694521858415765931064505193755202156521446156499075450033429983317127589636591133111239548821251790171694322930011927；

4、素性检测算法

实现 Milar-Rabin 素性检测算法，对于给定大数能够正确检测是否为素数，并且在实验报告中给出该算法的算法流程图及伪代码；并分析检测次数与能够正确检测的概率之间的关系；

测试点：1）1000023；

2）1000033；

3）100160063；

4）1500450271；

5）1494462659429290047815067355171411187560751791530；

6）22490812876539885046336053040043361022772062226905764414319531416752624982967181455912526153033030222985778230314070837549143068021815197910334221004333099

7）173114538715442253801652636578504897235814058376012019984132280493073144140873423822066926533851768593567972986030786930865304524765873917291156820356593465395949615668311730524585862713216977118030162614331116320577533153712280997129347743623082819252354000224098702300466561157715990374851814717133985999661

5、在给定不可约多项式的情况下， 编程实现上的加减乘运算， 以及 上一元多项式的带余除法运算；

样例： a) 0e \* 74 = ae; b) cf \* 8d = ea;

c) b9 / 74 = 03……25; d) cf / fc = 01……33;

测试点： a ）89 +/- 4d；

b）af +/- 3b；

c ）35 +/- c6；

d）ce \* f1；

e ）70 \* 99；

f）00 \* a4；

g）de / c6；

h）8c / 0a；

i）3e / a4；

6、在给定不可约多项式的情况下，编程实现上的快速模幂运算。

样例： a) 4d ^ 63108 = 63 b) 7b ^ 21902 = 3e

测试点： a) 89 ^ 18829；

b）3e ^ 28928；

c ）19 ^ 26460；

d）ba ^ 13563；

7、在给定不可约多项式的情况下，编程实现上的扩展欧几里得算法；

样例： a) exgcd(0e, 74) = (19, 03, 02); b) exgcd(b9, 65) = (2f, 7e, 01)

测试点：a ）exgcd(75,35)；

b）exgcd(ac,59);

c ）exgcd(f8,2e);

d）exgcd(48,99);

8、实现给定不可约多项式情况下，求 中元素的逆元；

样例： a) 3c ^{- 1} = 77 ; b) ea ^{- 1} = d7 ;

测试点：a ）8c；

b）be；

c ）01；

d）2d；

【选做】

1、 厄拉多塞筛法

实现厄拉多塞筛法并尝试进行优化，对于给定输入 N，请输出[2,N]之间所有素数；比较不同优化方法的时间、存储效率。

测试点：1）2；

2）103；

3）；

4）；

5）4275117753

2、 实现对 上的所有8次本原多项式的判定和生成。

样例： 1cf=111001111=x^8+x^7+x^6+x^3+x^2+x^1+1 属于 上的8次本原多项式；

提示：若 是 上的不可约多项式且的根是的本原元，则称为上的本原多项式； 另一等价定义如下：

若一个n次多项式 满足以下条件：

（1） 为不可约多项式；

（2） 可整除；

（3） 除不尽 ；

则称 为n次本原多项式。

【报告要求】

1、 实验布置两周内，请同学们将代码与实验报告分别提交至OJ平台对应位置，逾期者酌情扣分；命名格式为：学号\_姓名\_实验一；

2、 对于每个算法，报告中应含有函数调用关系图、测试样例及运行结果截图，部分算法需要流程图/伪代码；并且记录自己本次实验的收获感想，和对实验不足之处的建议；

3、 报告格式见附件二；且该格式仅供参考，同学们可酌情更改。

# 【思考题】

1、 思考厄拉多塞筛法中如何减少重复比较以提高速度；

2、 考虑中国剩余定理的实现中是否有可以改进的地方。

3、 在中，求逆运算非常复杂，思考是否有其他方法能够提供计算方式更简单的求逆算法。