Mathematics Explorers' Club Fall 2012
Number Theory and Cryptography

Chapter 0: Introduction

Number Theory enjoys a very long history – in short, *number theory is a study of integers*. Mathematicians over millennia study how different integers are related to each other. For example, they ask something like:

- Does the sequence 11, 111, 1111, 11111, … contain a square number?
- Are there non-zero integer solutions for the equation

  $$x^n + y^n = z^n$$

  for different values of n? (For example, when $n=2$, $(x,y,z) = (3,4,5)$ is a solution)
- Can every even number be written as sum of two primes?

These questions all sound good and abstract. Do we have a more practical use of the theory? This brings us to the other part of the module – Cryptography. There are a few parts in cryptography:

1) Substituting alphabets with integers, we can write any words or sentences in numbers. For example, we can assign A to 1, B to 2, C to 3 and so on.
2) Designing an algorithm to encode the numbers, and an `inverse algorithm' to decode the encoded numbers. The decoding algorithm is only known to the person who you wish to send your message to.
3) The numbers are encoded using the algorithm, which is a junk of random numbers to anyone but the person who owns the decoding algorithm.

And, of course, the design of such encoding and decoding algorithm relies heavily on the abstract number theory! In particular, we will learn one of the cryptography methods which is still heavily used on credit card transactions, internet shopping, etc. these days.

Chapter 1: Congruence

1.1 Basic Notions

We begin our Chapter with some simple questions. Try to find the odd one out for the following numbers:

a) 10, 15, 23, 165, 2000, 2170

b) 11, 16, 24, 166, 2001, 2171

c) 13, 21, 25, 39, 49, 65

d) 1, 11, 111, 1111, 11111, 111111

(a) is easy. 23 is not a multiple of 5 while the others are. With the similar mentality, we choose 24 in (b) since it is the only integer not ending with 1 or 6. Put it in another way, 24 is the only integer with remainder not equal to 1 when divided by 5.
For (c), we choose 39 out for the same reason as (b) – the other integers have remainder 1 when divided by 4, while 39 have a remainder of 3. And so does (d).

The above examples bring out the idea of congruence –

Let *m* be an integer. We say *a* is congruent to *b* modulo *m*, or

$$a \equiv b \pmod{m}$$

if *a*–*b* is a multiple of *m,* or if *a* and *b* have the same remainder upon dividing *m*.

For example, if a is a multiple of 9, then

$$a \equiv 0 \pmod{9}$$

Also, we have

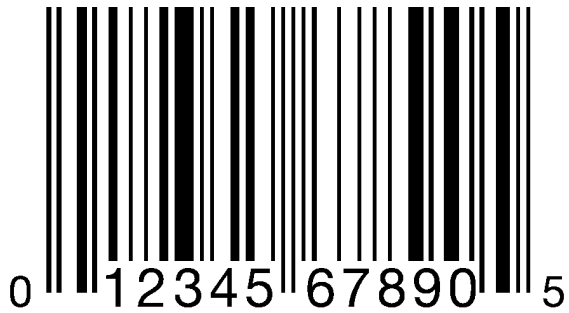
$$11 \equiv 16 \equiv 166 \equiv 2001 \equiv 2171 \pmod{5}$$

$$11 \equiv 111 \equiv 1111 \equiv 11111 \pmod{4}$$

1.2 Some Practical Examples

1) Check Digits

Nearly every item we purchase has a Universal Product Code (UPC). This is the string of numbers appearing next to the bar code of the product. A type of UPC, UPC-A, contains 12 digits. The first 6 indicate the manufacturer, the next 5 indicate the product and the last one (5 below) is a **check digit**.



A check digit is a redundant digit, which can be determined by the first 11 digits. With a check digit, one can detect simple errors in the input of a series of digits, such as a single mistyped digit or some permutations of two successive digits.

To determine the check digit we do the following. Denote digits 1 though 12 as $(a_1,a_2,a_3,...,a_{12})$ then compute

$$b \equiv (a_1,a_2,a_3,...,a_{11})*(3,1,3,1,3,1,3,1,3,1,3) \pmod{10}$$

Where * denotes the dot product, which is short hand for $3a_1+a_2+3a_3+...+3a_{11}$. Then

$$a_{12} = 10 - b$$

To check whether a UPC-A code is valid, we do the following:
Let's check whether 036000 291453 is a valid one. We compute

(0,3,6,0,0,0,2,9,1,4,5)*(3,1,3,1,3,1,3,1,3,1,3) mod 10

which is 58 (mod10) = 8. Then a_{12} must be 10-8 = 2. However, the 12^{th} digit is 3 above, so it cannot be a valid UPC-A code. There must be an error for the input.

The advantage of using this scheme is that it will **detect all errors involving one digit** and **nearly all errors involving the transposition of two adjacent digits**. If we switch the 2 and 3 positions in our example giving us a UPC of 063000 291452 the computation gives us 66mod10=6, not 0.

Problems:

1) Check whether the following UPC-A codes are valid or not:

Another example of check digit is the ten-digit International Standard Book Number (ISBN). The last digit of the ISBN is a check digit. This check digit system helps ensure that you are purchasing the correct text book. The way that the check digit works for ISBNs is comparable to the UPC system. To verify that the ISBN is correct, we do this computation.

$$(a_1,a_2,a_3,\ldots,a_9,a_{10})*(10,9,8,7,6,5,4,3,2,1) \pmod{11}$$

If this does not yield 0, then there is an error in your ISBN.

Sometimes it is necessary for the last digit to be 10, in this case the ISBN will end with the letter X. The check digit will always detect if you wrote one of the numbers down incorrectly.

2) <u>Caesar's Cipher</u>

This is the first and easiest encryption method. To start encrypting our message, we first convert our alphabet system into numbers in the obvious way – A to 1, B to 2 and so on. We have the following table of conversion:

Conversion Table

A	= 1	K	= 11	U	= 21
B	= 2	L	= 12	V	= 22
C	= 3	M	= 13	W	= 23
D	= 4	N	= 14	X	= 24
E	= 5	O	= 15	Y	= 25
F	= 6	P	= 16	Z	= 26
G	= 7	Q	= 17		
H	= 8	R	= 18		
I	= 9	S	= 19		
J	= 10	T	= 20		

For example, JAMES will be converted to "10,1,13,5,19". To start encoding the numbers, we do the following algorithm to every letter in the message:

x goes to x + 8 (mod 26)

Therefore, the encoded word JAMES will read "18,9,21,13,1". For those who do not know the encoding, the bunch of number reads SIUMA, which makes no sense at all.

As we mentioned at the beginning, we need an algorithm to 'decode' the message, and it is given by

y goes to y - 8 (mod 26)

For example, to decode the message “21,9,26,7”, we subtract 8 to each number, having “13,1,18,-1”. However, $-1 \equiv 25 \pmod{26}$, so the decoded message is “13,1,18,25”, which is MARY according to the table.

Problem:

Discuss some of the disadvantages of Caesar’s Cipher (from the perspective of a student some 2000 years after Caesar).

Chapter 2: Solving Congruence Equations I

To better utilize the power of congruence, one shall learn how to solve equations involving congruence. We first start with the linear congruence equations

$$ax \equiv b \pmod{n}$$

2.1 Some Simple Examples

Example 1 – Solve $x + 3 \equiv 13 \pmod{17}$

Solution:

This is easy. We can take away 3 both sides to get $x \equiv 10 \pmod{17}$.

Note that it means there are **MANY** solutions to the equation. In fact, any integer that has remainder 10 upon dividing 17 are solutions to the equation!

Example 2 – Solve $x + 25 \equiv 3 \pmod{6}$

Solution:

We can do the same thing as in Example 1, getting $x \equiv -22 \pmod{6}$. But remember $0 \equiv 24 \pmod{6}$. Hence, by property (1) we can add 0 on L.H.S., and 24 on R.H.S., yielding $x \equiv 2 \pmod{6}$.

Example 3 – Solve $5x \equiv 2 \pmod{13}$

Solution:

Remember we need x to be an integer, so $x = 0.4$ is absurd. Think of an integer a so that $5 \times a \equiv 1 \pmod{13}$. For example in our case $a = 8$ since $5 \times 8 = 40 = 3 \times 13 + 1$. After getting such a, we multiply it on both sides of the equation, getting

$$\begin{aligned} 5x \times 8 &\equiv 2 \times 8 \pmod{13} \\ 40x &\equiv 16 \pmod{13} \\ 40x \equiv x &\equiv 16 \pmod{13} \\ x &\equiv 16 \equiv 3 \pmod{13} \\ x &\equiv 3 \pmod{13} \end{aligned}$$

Problems:

Solve the following congruence equations (if there are no solutions, give a brief reason why it is so)

a) $x + 14 \equiv 2 \pmod{15}$

b) $4x \equiv 7 \pmod{23}$ [Hint: Note that $24 \equiv 1 \pmod{23}$]

c) $3x + 9 \equiv 21 \pmod{39}$

d) $3x + 9 \equiv 2 \pmod{39}$

e) $8x + 3 \equiv 16 \pmod{17}$

f) $26x + 3 \equiv 1 \pmod{5}$

2.2 Solving More Difficult Equations – Euclidean Algorithm

Our equations so far dealt with some small numbers, one may ask in general, how do we solve equations like

$$329x + 143 \equiv 16 \pmod{271}$$

Inspired by Example 3 above, if we can find an integer a so that $329a \equiv 1 \pmod{271}$, then we can proceed by multiplying both sides by a:

$$a(329x + 143) \equiv 16a \pmod{271}$$

$$a(329)x + 143a \equiv 16a \pmod{271}$$

$$a(329)x \equiv 1.x \equiv 16a - 143a \pmod{271}$$

The problem is, **how to find such *a* in general? Or does such *a* exist at all?**
The answer is given by Euclid, a Greek Mathematician in 3rd century BC! Here is the statement of what Euclid proved in his book of 'Elements':

> **Theorem (Euclidean Algorithm):**
> Suppose the greatest common divisor of p and q, denoted (p,q) is r.
> Then there are integers a and b so that
> $$ap + bq = r$$
> Moreover, there is an algorithm determining the values of a and b.

To put the theorem into perspective, take $p = 329$, $q = 271$. Euclid's algorithm tells us two things:
1) the g.c.d. of 329 and 271 (which is 1 in this case).
2) the numbers a and b satisfying $329a + 271b = 1$, so $329a \equiv 1 \pmod{271}$!

I will roughly show the algorithm below:
329 = 1 x **271** + **58**
271 = 4 x **58** + **39**
58 = 1 x **39** + **19**
39 = 2 x **19** + **1**
19 = 1 x 19
And since we cannot proceed any more, the 1 appearing in the second last equality is the g.c.d. of 329 and 271.

To find out the values of a and b, we just need to "substitute backwards":
329 – 1 x 271 = 58
271 = 4 x (329 – 1 x 271) + 39
329 – 1x271 = 1 x [271 – 4 x (329 – 1 x 271)] + 19
271 – 4 x (329 – 1 x 271) = 2 x {329 – 1x271 – 1 x [271 - 4 x (329 – 1 x 271)]} + 1
Therefore,
5 x 271 – 4 x 329 = 2 x {329 – 1x271 – 271 + 4 x (329 – 271)} + 1
5 x 271 – 4 x 329 = 2 x {5 x 329 – 6x271} + 1
5 x 271 – 4 x 329 = 10 x 329 – 12x271 + 1
-14 x 329 + 17 x 271 = 1

Problems:
1) Find the g.c.d. of the following pairs of integers, and find the a and b as above:
(a) (245, 154)

(b) (187, 323)

2) Solve the congruence equation

$$154x \equiv 21 \pmod{245}$$

$$187x \equiv 16 \pmod{323}$$

3) Can you give a criterion on whether the equation

$$px \equiv n \pmod{q}$$

has solutions, in terms of n and $r = (p,q)$?

Chapter 3: Solving Congruence Equation II

As we move forward to studying the RSA encryption, we often need to solve equations of the form

$$a^x \equiv 1 \pmod{n}$$

For example,

$$2^x \equiv 1 \pmod{7}$$

has a solution x = 3.

Problem:

1) The list of the powers of 2 is given as follows:

1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096, …

Try to find their remainders upon dividing 7, and list the remainder below:

Did you see a pattern? What other values of *x* will satisfy $2^x \equiv 1 \pmod{7}$?

However, not all choices of a and n will have a solution. For example

$$3^x \equiv 1 \pmod{15}$$

does not have any solutions. (Why?)

So the questions are, again, the following:

- When does the equation have solutions?
- If it has solutions, what are the solutions?

3.1 Euler ϕ–function

It turns out that the (partial) answers to the above questions rely a lot on the φ-function, which is introduced by Leonhard Euler, a Swiss mathematician in the 1700s. Here is the definition of the φ-function:

$\varphi(n)$ is the number of positive integers a with $(a,n) = 1$.

Where (a,n) means the greatest common divisor of a and n, e.g. (12, 18) = 6.

Problems:

2) Let $n = 5$, list all the integers a such that $(a,5) = 1$. How many possible values of a are there? The number of possible values of a is $\varphi(5)$.

   Do the same for $n = 7, 13, 23$. What are $\varphi(7)$, $\varphi(13)$ and $\varphi(23)$?

   Can you give a formula for $\varphi(p)$ if p is a prime number?

3) We have seen how $\varphi(p)$ can be computed if p is prime. Now let's try a little bit harder. Let $n = 3, 9, 27, 81$, list all integers a such that $(a, n) = 1$. Or you can go straight ahead to find $\varphi(n)$ for $n = 3, 9, 27, 81$.

   Do you see a pattern? What is $\varphi(729)$?

4) Now let's try to do the same for composite number of two different primes. Find $\varphi(15)$, $\varphi(21)$, $\varphi(35)$.

   What is the relationship between $\varphi(35)$, $\varphi(5)$ and $\varphi(7)$?

We have actually figured out the following rules:

- For k a positive integer, and p prime, we have the formula

$$\varphi(p^k)=p^{k-1}(p-1)$$

- The function φ is *multiplicative*, which means if $(a,b) = 1$, then

$$\varphi(ab)=\varphi(a)\varphi(b)$$

3.2 Euler Theorem

Now we are relating our original question of solving equations to the φ-function we just explored:

> **Theorem (Euler's Theorem):** For all integer n, and for all a so that $(a,n) = 1$,
>
> $$a^{\varphi(n)} \equiv 1 \pmod{n}$$

Let's put the theorem in practice!

Problems:

5) Let $n = 5$. We have seen from last page that $\varphi(5)=4$. So the theorem says

$$a^4 \equiv 1 \pmod{5}$$

for any a satisfying $(a,5) = 1$. So let's try to compute

$2^4 \pmod 5$, $3^4 \pmod 5$, $4^4 \pmod 5$, $6^4 \pmod 5$

What about $5^4 \pmod 5$? Why is it not 1(mod 5)?

6) Can you immediately tell the remainder of 673^{268} upon dividing 269?
(Hint: 269 is a prime number!)

7*) Can you immediately tell the remainder of 31^{18} upon dividing 7?

Problems:

8*) This is an important idea when we do RSA encryption next week. Let n = 15. Then $\varphi(15)=2x4 = 8.$ So

$$a^8 \equiv 1 \pmod{15}$$

if a is not divisible by 3 or 5. By using the powers of 2 on the first page, try to check it is true for a = 2, i.e. is $2^8 \equiv 1 \pmod{15}$?

If $2^8 \equiv 1 \pmod{15}$, we have seen that $2^{16} \equiv 1 \pmod{15}$, $2^{24} \equiv 1 \pmod{15}$ and so on. How about $2^9 \pmod{15}$, $2^{17} \pmod{15}$, $2^{25} \pmod{15}$ and so on?

How about $2^{10} \pmod{15}$, $2^{18} \pmod{15}$, $2^{26} \pmod{15}$ and so on?

9**) Can you immediately tell the remainder of 31^{19} upon dividing 7?

Chapter 4 - RSA Cryptography

We finally come to the climax of the course – RSA Cryptopgraphy! RSA got its name from the last initials of the three people that first publicly described it in 1977, Ron Rivest, Adi Shamir, and Leonard Adleman, who were at MIT. RSA is very widely used in electronic commerce protocols, and is believed to be secure given sufficiently long keys combined with up-to-date implementations.

Before we move on, let's recall the crucial observation we made last week:

1) Pick a product of two prime numbers $n = pq$,
2) The Euler function $\varphi(pq)=(p-1)(q-1)$
3) Therefore, Euler's Theorem says if a is not a multiple of p or q,
$$a^{(p-1)(q-1)} \equiv 1 \pmod{pq}$$
4) So $x = (p-1)(q-1)$ is a solution of the equation
$$a^x \equiv 1 \pmod{pq}$$
5) There are more solutions, namely all the multiples of $(p-1)(q-1)$ are solutions.
6) Here is the gist:
IF WE KNOW $a^x \equiv 1 \pmod{pq}$, **THEN** $a^{x+1} \equiv a \pmod{pq}$
So $a^{\text{"a multiple of (p-1)(q-1)"} + 1}$ is going to have remainder a upon dividing over pq

Now let's try to put RSA into practice:

How RSA works:

1) Pick p = 31 and q = 29, so pq = 899, and $\varphi(899)=(29-1)(31-1)= 840.$
We can now encode any number between 1 and 26!
Say we want to encode 5. By Euler's theorem we know
$$5^{840} \equiv 1 \pmod{899}$$
And also $5^{2x840} \equiv 1 \pmod{899}$, $5^{3x840} \equiv 1 \pmod{899}$, $5^{4x840} \equiv 1 \pmod{899}$…
Hence $5^{2x840+1} \equiv 5 \pmod{899}$, $5^{3x840+1} \equiv 5 \pmod{899}$, $5^{4x840+1} \equiv 5 \pmod{899}$…

Now pick e=31. This is the encryption key – we encrypt our number 5 by taking
$$5^e \pmod{899}$$
I get $5^{31} \equiv 67 \pmod{899}$. So the encrypted number is 67.

To decrypt 180, we need to find a number d so that de= "multiple of 840" + 1
But yes, d = 271 gives 271 x 31 = 10 x 840 + 1

Therefore

$$5^{31x271} \equiv 5^{10 \times 840 + 1} \equiv 5 \pmod{899}$$

Going back to decryption, with the encrypted number 67, we just need to take

$$67^{271} \equiv (5^{31})^{271} \equiv 5^{31x271} \equiv 5 \pmod{899}$$

Now let's see how the above mechanism works in practice:

1) Alice wants to send a secret letter A,B,…,Z to Ben
2) Ben comes up with the number n=899, e=31, d=271.
3) Ben announces the numbers (n,e)=(899,31) publicly. This is called **PUBLIC KEY**.
4) The secret letter Alice has in her mind is "M", which is translated into a number 13.
5) With the public key, Alice encrypts her message by taking

$$13^{e} \pmod{n} = 13^{31} \pmod{899}$$

6) The encrypted message is 602. No one knows how to decrypt 602 except Ben.
7) To decrypt, Ben only needs to do

$$602^{271} \pmod{899}$$

Which is precisely 13, the message Alice sent out.

Now let's play with the numbers!

1) (n, e) = (1147, 29) d = ??
2) (n, e) = (1763, 71) d = ??
3) (n, e) = (3127, 431) d = ??

The above numbers are good enough to encode numbers from 1 to 26. So one may ask, how about bigger numbers? This is simple, we just need to pick two big prime numbers *p* and *q* to produce *n*!

In order to break the code, one needs to know

a) How *n* is factorized into two primes *p* and *q*
b) After knowing what p and q are, we find *d* so that *de* is "a multiple of $(p-1)(q-1)$" + 1. This can be done using the Euclidean algorithm.

Remember the money awards (up to $50,000!) on factorizing *n* with 200+ digits? Now you know why it worth that much money!

Finale: **Make your own RSA code!**

With the aid of a computer, you can make your own RSA encryption! Here are the steps:

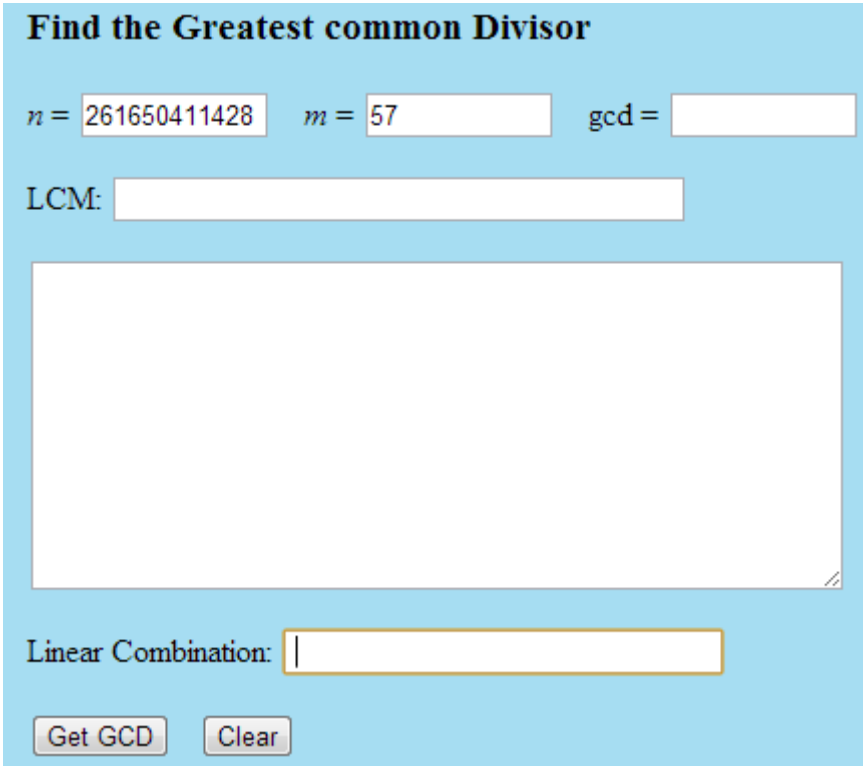
Step 1: Pick two BIG prime numbers. This can be done in the website below-
http://markknowsnothing.com/cgi-bin/primes.php
Type a random number, and the website will show you all the primes close enough to the number you picked. As an example, my choice of primes are p=57947 and q=4515419. Hence my n is

$$n = pq = 57947 \times 4515419 = 261654984793$$

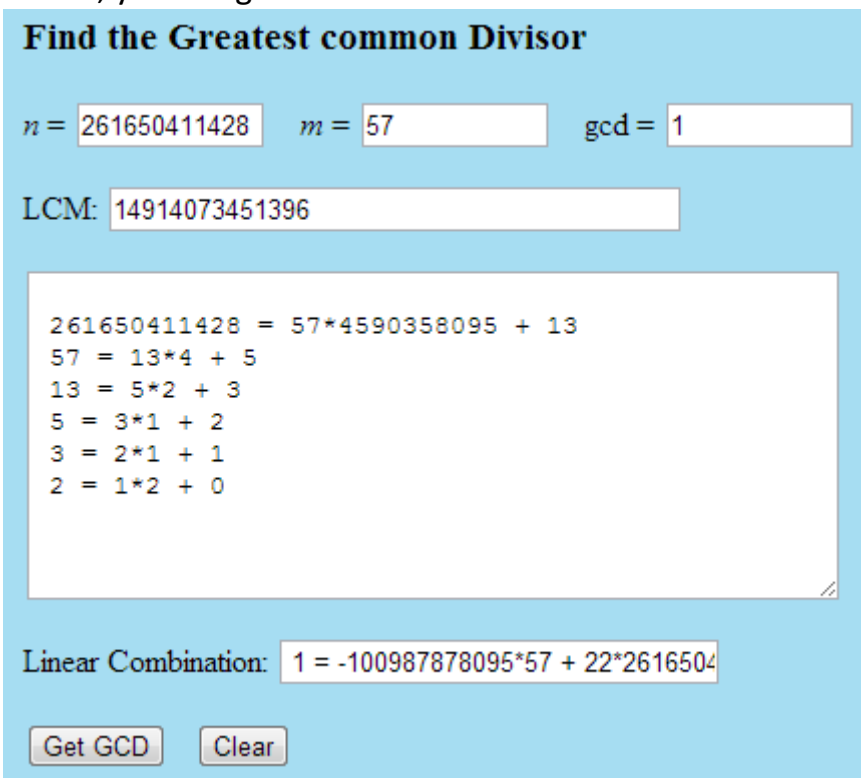
I can now encode almost every number below n.

Step 2: Pick the 'encryption number' e. Generally any odd number smaller than n works. I pick e = 57 as an example.

Step 3: To see whether e = 57 is a good choice, use the following website:
http://www.math.sc.edu/~sumner/numbertheory/euclidean/euclidean.html
Put the "n" slot with value (p-1)(q-1) = 261650411428, and "m" slot with e = 57 as shown below:



Click “get GCD”, you will get



e = 57 is a good choice as long as the GCD is 1.

Step 4: Now find the decryption number d. It is shown in the bottom line of the calculation, right next to the number 57. In our example,

$$d = -100987878095$$

But wait, we don’t want a negative power of d, so we add *d* = -100987878095 by 261650411428 and get

$$d = -100987878095+261650411428 = 160662533333$$

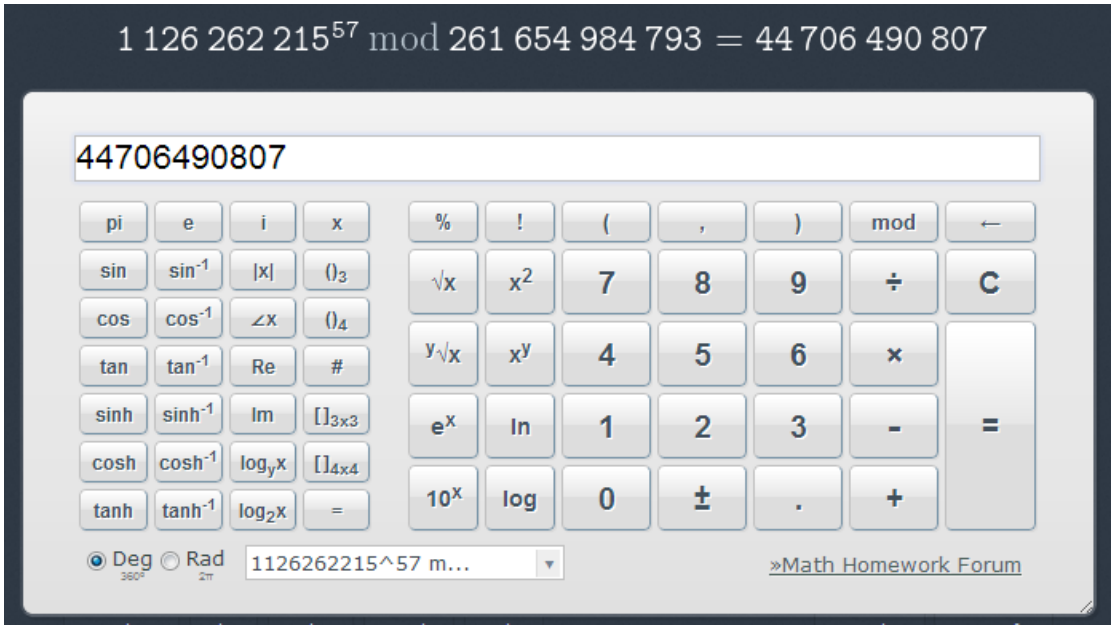
KEEP THIS NUMBER PRIVATE!

Step 5: Announce (*n*,*e*) to the public, and convert your message using the table below

Conversion Table

A = 11	K = 21	U = 31
B = 12	L = 22	V = 32
C = 13	M = 23	W = 33
D = 14	N = 24	X = 34
E = 15	O = 25	Y = 35
F = 16	P = 26	Z = 36
G = 17	Q = 27	
H = 18	R = 28	
I = 19	S = 29	
J = 20	T = 30	

With the conversion table, convert APPLE into 1126262215. Now encode APPLE by $1126262215^{e} \pmod{n}$, using the website below –
http://web2.0calc.com/



The encoded message is 44706490807.

Step 5: To decode the message, we just need to do $44706490807^{d} \pmod{n}$ (Be careful that the online calculator cannot deal with very big powers!)

We get back the number 1126262215!