Results 1 to 10 of 17

THREAD:
NEW ONE WAY FUNCTION

03-11-2013, 03:52 PM #1

trurl_
Registered User

Join Date: Dec 2004
Posts: 133

NEW ONE WAY FUNCTION

Think of 2 clocks one going backwards losing 10 minutes every hour at 2:15 pm and another clock gaining 7 minutes every hour at 7:21. Will these clocks ever have the same time in b iterations of the clock changing? They should possibly if there where infinity hours and infinity iterations.

So instead of a clock, why not take the Mod of a decimal? (Yes I know we use Mod for whole numbers). But how many iterations would it take of 2 decimals each the result of division into a whole number to meet?

This is more computational intensive than Prime numbers.

http://www.constructorscorner.net/Fi...ayFunction.pdf

It's all about ideas.

constructorscorner.com

Blog this Post | Edit Post | Reply | Reply With Quote

03-15-2013, 09:13 AM #2

mmakrzem
Game Developer

Join Date: Mar 2005
Location: Ontario, Canada
Posts: 1,719

Is this what you want: http://php.net/manual/en/function.fmod.php

C++, 3D OpenGL and Game Programming video tutorials:
www.MarekKnows.com
Play my free games: Ghost Toast, Zing, Jewel Thief

Blog this Post | Reply | Reply With Quote

03-20-2013, 02:46 AM #3

trurl_
Registered User

Join Date: Dec 2004
Posts: 133

Thanks for the link on programming Modulus. I really haven't studied that in detail yet. I know the basics of it relating to the revolutions and having full revolutions. I am expanding upon that to decimals. I know that doesn't make sense. But I want to find a pattern that shows when the clock is approaching a revolution.

This is just an idea. It needs concrete work. But I thought if you had a decimal number and could find when the decimal number equaled a whole number there would be a one way function. You would have a product without an easy way to decide which decimal formed it.

The approach is simplistic. I don't even try to claim that I understand cipher schemes, but as RSA uses 2 numbers to form a product that can't easily be reversed I see potential for decimals to equal whole numbers. There is potential to use this decimal equals a whole number.

However the goal is not only to find a one way function, but the pattern of division; to extend modulus to decimals; not an easy thing. But what if there was a way to say that the modulus of a number by a given number is approaching zero? You cannot do this yet because different numbers have different multiples. But I am saying a pattern is possible, it just needs uncovered.

I am saying this is worth research based on intuition. I see this working I just can't put it into words. Does anyone see any merit in this theory?

Blog this Post | Edit Post | Reply | Reply With Quote

04-30-2013, 01:21 AM #4

trurl_
Registered User

Join Date: Dec 2004
Posts: 133

Now that Csans has proven that rational numbers are represented by fractions. It is time for a little numerology or maybe a theory to test. **If a decimal is repeating or terminating than it must have a rational number representation.**

I believe this rational number fraction could be used as a one way function just as Prime numbers are. There is certainly easier to choose a rational number set. But can it be as intensive as factoring 2 Prime number's product. I don't know. After all I don't really understand how to use the one way function once you find one. I am just studying this myself now. But I feel there is something here and it is worth a short inspection. Of course it is just a theory and I can easily be completely wrong. But if you were interested in the thread on irrationals, this is a more worthy problem to pursue.

Blog this Post | Edit Post | Reply | Reply With Quote

04-30-2013, 08:14 AM #5

ComicSansMS
Most Horrible Font Ever

Join Date: Jun 2003
Location: Trier, Germany
Posts: 1,328

> Originally Posted by **trurl_**
> *I believe this rational number fraction could be used as a one way function just as Prime numbers are.*

Probably not. Computing the fraction from a rational given in decimal notation is equivalent to finding the greatest common divisor (this is 7th grade math), which can be done efficiently using Euclid's algorithm. The whole point about using primes in RSA is that integer factorization is (believed to be) non-polynomial.

And you also must not forget the another crucial property that is often overlooked: Factorization is hard, but testing for primality *is not*. An algorithm where key generation is just as hard as cracking would be quite worthless.

A Pageant of the Bizarre

Blog this Post | Reply | Reply With Quote

05-20-2013, 05:36 AM #6

trurl_
Registered User

Join Date: Dec 2004
Posts: 133

I get "this is seventh grade math" a lot. But I did think about the possibility of a one way function with some thought. I would like to think of it as a hypothesis more than numerology even though numerology does have geometry and calculations. There has been much geometry from those trying to draw magic or find divinity with symbols. Part of math is trying to unlock secrets. That is not the way schools teach, but there is much to be learned there. I'm not into numerology, but to those that are there is nothing wrong to search for answers even though those are not my religious beliefs and I am getting off topic.

You discount my idea because it is only an idea. I have to give you reason to pursue a further inspection. Here is my argument:
I don't know how the enciphering of encryption is done. Obviously the keys are in some ways independent of the keys (I think). With RSA the public key must encipher the document so that knowledge of the public key does not allow the public key to decipher. This again depends on the enciphering algorithm. Even with a one way function knowledge of one key would allow the document to be decoded. This is the part that is really hard to understand.

But here is my proposal:

You need a 700 digit number that was found by the private keys of a large decimal and multiple that when multiplied by the decimal equals that large digit number. The large number is the public key. The private keys are the decimal number and the multiple (or number of iterations of the decimal + itself).

The security comes by the fact that many numbers can divide into the 700 digit number, but with a decimal value it is hard to test which decimal was used. Having a complex decimal number whose iterations of adding itself to equal a whole number would be difficult to find. It would be even harder than Prime numbers. Trying to solve this would lead to many inconclusive answers.

But this all depends on how messages are enciphered with the public key. I do not know this. I am mostly basing this on a one way function. I am probably wrong, but I just want to show that my idea was not numerology. It seems too simplistic to work or not to be tried before. This is the similarities I saw in the irrational number post.

It is also important to note that Whit Diffe had a one way function recommended to him by a former military cryptographer. It is in the book "Crypto" by Steven Levy. I can't find the scheme he recommended in the book yet, but it was not Prime numbers like most of today's crypto.

Also, the British Cryptographer claims to have invented RSA before MIT. I don't believe him.

I know all one way functions do not equal encryption, but if anyone understands the enciphering algorithm part: what is the reason my keys won't work?

It's all about ideas.

constructorscorner.com

06-17-2013, 03:16 AM #7

trurl_
Registered User

Join Date: Dec 2004
Posts: 133

Don't take my uneducated opinion but this works for 85; two test values 3 and 5 (because 3 is small).

This is a Java program. Just change the value of product from 85 to another product.

Post what you think.

Code:

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package rsatruth;

//import java.util.Scanner;

/**
 *
 * @author Trurl
 */
public class RSATruth {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // ((85^4/x + 2 *(85^2 * x^2) + x^5)/85^3)* x - 85

        double product = 85, answer;
        int x;
        double key;
        //System.out.print("Enter the Product of to Prime numbers  ");
        //product = keyboard.nextDouble();

        for (x = 3; x < product; x +=2)
        {
        answer = ((((Math.pow(product,4)/x) + 2 * (Math.pow(product,2) * Math.pow(x,2))) / (Math.pow(product,3))) * x -
85);
```

It's all about ideas.

constructorscorner.com

06-25-2013, 10:36 PM #8

Join Date: Dec 2004

trurl_
Registered User
Posts: 133

UPDATED WITH SIMPLER ESTIMATE.

Here is a simpler estimate of N of RSA. Let me know if it helps or is completely wrong. I found it helpful in estimating. However solving the polynomial itself is difficult.

Code:

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package rsatest20130621;

/**
 *
 * @author Trurl
 */
public class Rsatest20130621 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {


        double product = 85, answer;
        int x;
        //double key;
        //System.out.print("Enter the Product of to Prime numbers  ");
        //product = keyboard.nextDouble();

        for (x = 3; x < product; x +=2)
        {

            //answer = (((Math.pow(product,4)/x) / ((Math.pow(product,4))/((Math.pow(x,4))) + 2 * ((Math.pow(product,2)/x) +
Math.pow(product,2)* x) + (Math.pow(x,4)))) - x   );
            //answer = (( ((Math.pow(product,2)) + (Math.pow(x,3)) ) / product) - product ) ;
```

It's all about ideas.

constructorscorner.com

Blog this Post | Edit Post | Reply | Reply With Quote

09-27-2013, 01:39 AM #9

trurl_
Registered User
Join Date: Dec 2004
Posts: 133

Don't believe my uneducated math. Call it numerology. But the code doesn't lie.

RSA was never meant to be a one way function. It is a sham. It made the user believe the message is protected by a pattern of Primes. But finding N is just a matter of substitution. Finding the pattern in the process is simple and not protected by a pattern of Prime numbers!

Code:

```
y = ((85/x) * 85 - x^2)/ x = ((85^2/x) + x^2)/ 85

In[5]:=
p = ((85/x) * 85 - x^2)/ x  - ( ((85^2/x) + x^2)/ 85);

sol = NSolve[p == 0, x]

Out[6]= {{x -> -86.893}, {x -> -7.50438 +
    19.0222 I}, {x -> -7.50438 - 19.0222 I}, {x -> 16.9017}}
```

It's all about ideas.

constructorscorner.com

Blog this Post | Edit Post | Reply | Reply With Quote

09-28-2013, 02:51 AM #10

trurl_
Registered User

Join Date: Dec 2004
Posts: 133

IMPORTANT CORRECTION

Code:

```
y = ((85/x) * 85 - x^2)/ x = ((85^2/x) + x^2)/ 85

In[5]:=
p = ((85/x) * 85 - x^2)/ x  * ( 85^2/((85^2/x) + x^2)) - 85;

sol = NSolve[p == 0, x]

Out[6]= {{x -> -86.893}, {x -> -7.50438 +
    19.0222 I}, {x -> -7.50438 - 19.0222 I}, {x -> 16.9017}}
```

It's all about ideas.

constructorscorner.com

Blog this Post | Edit Post | Reply | Reply With Quote

+ Reply to Thread

Page 1 of 2 1 2 Last

QUICK REPLY

Font Size

Show your signature

Post Quick Reply | Go Advanced | Cancel

« Previous Thread | Next Thread »

TAGS FOR THIS THREAD

None
View Tag Cloud

Add / Edit Tags

POSTING PERMISSIONS

You may post new threads
You may post replies
You may post attachments
You may edit your posts

BB code is On
Smilies are On
[IMG] code is On
[VIDEO] code is On
HTML code is Off

Forum Rules