JOURNEY INTO CRYPTOGRAPHY

Modular arithmetic

Fast modular exponentiation

Fast Modular Exponentiation

Modular inverses

The Euclidean Algorithm

NEXT SECTION:
Primality test

Fast modular exponentiation

How can we calculate A^B mod C quickly if B is a power of 2 ?

Using modular multiplication rules:

i.e. **A^2** mod C = **(A * A)** mod C = (**(A mod C)** * **(A mod C)**) mod C

We can use this to calculate 7^256 mod 13 **quickly**

7^1 mod 13 = **7**

7^2 mod 13 = (**7^1** ***7^1**) mod 13 = (**7^1 mod 13** * **7^1 mod 13**) mod 13

We can substitute our previous result for **7^1 mod 13** into this equation.

7^2 mod 13 = (**7** ***7**) mod 13 = **49** mod 13 = **10**

7^2 mod 13 = **10**

7^4 mod 13 = (**7^2** ***7^2**) mod 13 = (**7^2 mod 13** * **7^2 mod 13**) mod 13

We can substitute our previous result for **7^2 mod 13** into this equation.

7^4 mod 13 = (**10** * **10**) mod 13 = **100** mod 13 = **9**

7^4 mod 13 = **9**

7^8 mod 13 = (**7^4** * **7^4**) mod 13 = (**7^4 mod 13** * **7^4 mod 13**) mod 13

We can substitute our previous result for **7^4 mod 13** into this equation.

7^8 mod 13 = (**9** * **9**) mod 13 = **81** mod 13 = **3**

7^8 mod 13 = **3**

We continue in this manner, substituting previous results into our equations.

...after 5 iterations we hit:

7^256 mod 13 = (**7^128** * **7^128**) mod 13 = (**7^128 mod 13** * **7^128 mod 13**) mod 13

7^**256** mod 13 = (**3** * **3**) mod 13 = **9** mod 13 = **9**

7^256 mod 13 = **9**

This has given us a method to calculate **A^B mod C** quickly provided that **B is a power of 2**.

However, we also need a method for fast modular exponentiation when **B is not a power of 2**.

How can we calculate A^B mod C quickly for any B ?

i.e. $5^{117} \bmod 19$

Step 1: Divide B into powers of 2 by writing it in binary

$$117 = 1110101 \text{ in binary}$$

Start at the rightmost digit, let k=0 **and** for **each digit**:

- **If the digit is 1, we need a part for 2^k, otherwise we do not**
- **Add 1 to k, and move left to the next digit**

$$117 = (2^0 + 2^2 + 2^4 + 2^5 + 2^6)$$
$$117 = 1 + 4 + 16 + 32 + 64$$

$$5^{117} \bmod 19 = 5^{(1 + 4 + 16 + 32 + 64)} \bmod 19$$

$$5^{117} \bmod 19 = (5^{1} * 5^{4} * 5^{16} * 5^{32} * 5^{64}) \bmod 19$$

Step 2: Calculate mod C of the powers of two ≤ B

5^1 mod 19 = **5**

5^2 mod 19 = (**5^1** * **5^1**) mod 19 = (**5^1 mod 19** * **5^1 mod 19**) mod 19

5^2 mod 19 = (**5** * **5**) mod 19 = **25** mod 19

5^2 mod 19 = **6**

5^4 mod 19 = (**5^2** * **5^2**) mod 19 = (**5^2 mod 19** * **5^2 mod 19**) mod 19

5^4 mod 19 = (**6** * **6**) mod 19 = **36** mod 19

5^4 mod 19 = **17**

5^8 mod 19 = (**5^4** * **5^4**) mod 19 = (**5^4 mod 19** * **5^4 mod 19**) mod 19

5^8 mod 19 = (**17** * **17**) mod 19 = **289** mod 19

5^8 mod 19 = **4**

5^16 mod 19 = (**5^8** * **5^8**) mod 19 = (**5^8 mod 19** * **5^8 mod 19**) mod 19

5^16 mod 19 = (**4** * **4**) mod 19 = **16** mod 19

5^16 mod 19 = **16**

5^32 mod 19 = (**5^16** * **5^16**) mod 19 = (**5^16 mod 19** * **5^16 mod 19**) mod 19

5^32 mod 19 = (**16** * **16**) mod 19 = **256** mod 19

5^32 mod 19 = **9**

5^64 mod 19 = (**5^32** * **5^32**) mod 19 = (**5^32 mod 19** * **5^32 mod 19**) mod 19

5^64 mod 19 = (**9** * **9**) mod 19 = **81** mod 19

5^64 mod 19 = **5**

Step 3: Use modular multiplication properties to combine the calculated mod C values

5^117 mod 19 = (**5^1** * **5^4** * **5^16** * **5^32** * **5^64**) mod 19

5^117 mod 19 = (**5^1 mod 19** * **5^4 mod 19** * **5^16 mod 19** * **5^32 mod 19** * **5^64 mod 19**) mod 19

5^117 mod 19 = (**5** * **17** * **16** * **9** * **5**) mod 19

5^117 mod 19 = **61200** mod 19 = **1**

5^117 mod 19 = **1**

Notes:

More optimization techniques exist, but are outside the scope of this article. It should be noted that when we perform modular exponentiation in cryptography, it is not unusual to use exponents for B > 1000 bits.

Ask a question...

Questions Tips & Thanks Report a mistake

Guidelines

Top Recent

Is there a tutorial/challenge for writing numbers in binary? If not that would be a great addition!

15 Votes • 1 Comment • Flag 2 years ago by JMGClark

This link no longer works. Is there another video perhaps?

4 Votes • Comment • Flag 10 months ago by courtpope

Answer this question...

what if the exponent has four digits

4 Votes • Comment • Flag 2 years ago by Ismail Siddiqui

Modular exponentiation works for any exponent, even ones with 4 digits. Does this answer your question? If not, please be more specific.

EDIT: OK, I understand now. You just need an example. The process is the same, though.
OK, we need to figure out 3^1993 (mod 17).
1993-->Binary
The biggest power of 2 less than or equal to 1993 is 1024.
1993-1024
969
The biggest power of 2 less than or equal to 969 is 512.
969-512
457
The biggest power of 2 less than or equal to 457 is 256.
457-256
201
The biggest power... (more)

9 Votes • 4 Comments • Flag
2 years ago by The4thdimentionpro

Show all 2 answers • Answer this question

Why isn't this math on the learning dashboard, say in the world of math mission?

8 Votes • Comment • Flag about a year ago by Dana Wright

Answer this question...

I'm using a different method to calculate 7^256 mod 13 by dividing it into 7^3 and 7^253 then continuously diving the 7^3 into 7^253. I got the last iteration to be 7^4 which can be broken up into 7^3 and 7. And since 7^3 mod 13 is congruent to 1, that just leaves me with 7mod 13. What am I doing wrong!?

2 Votes • 1 Comment • Flag about a year ago by Mike G

7^3 mod 13 is congruent to 5

1 Vote • Comment • Flag

about a year ago by gunwati.rules

Answer this question...

Could one take the 5^117mod19=(5*17*16*9*5)mod 19 and make it equal to ((5*17)mod19)*(16*9)mod19*5mod19)mod19?

2 Votes • Comment • Flag 8 months ago by Jon Xu

If I'm not mistaken Khan Academy has been using the following in the lessons without proof. The result for multiplication supports what you want to do. My attempt at proof is included here in case you're interested

multiplication: (N1*N2*...)modC = (N1modC*N2modC*...)modC for any number of factors.
addition: (N1+N2+...)modC = (N1modC+N2modC+...)modC for any series of numbers summed together.

Proof for multiplication:
given: A*B modC = (A modC * B modC)modC.
prove: E*F*G modC = (E modC * F modC * G... (more)

1 Vote • Comment • Flag 7 months ago by Joe Mason

Answer this question...

What is the remainder when 2^1990 is divided by 1990?

1 Vote • Comment • Flag 2 years ago by Jojo Rkmv

2^1990 mod 1990 = 1024

If you apply the technique of fast modular exponentiation you should be able to calculate that. If you plug the numbers into this program https://www.khanacademy.org/math/applied-math/cryptography/modarithmetic/p/fast-modular-exponentiation

it will illustrate the steps

3 Votes • Comment • Flag 2 years ago by Cameron

Answer this question...

What is iterations?

1 Vote • Comment • Flag

6 months ago by CommanderCT911 Of Recon Corps

the repetition of a process or utterance.

1 Vote • 1 Comment • Flag 6 months ago by tware6119

Answer this question...

so, does any power of b require binary computer language? isn't there any other way out?

1 Vote • Comment • Flag 10 months ago by Pulkit Gopalani

Answer this question...

Can anyone tell how to convert digits into binary?

1 Vote • Comment • Flag 2 months ago by Pranshu The Great

Here's the lessons for converting between bases:
https://www.khanacademy.org/math/pre-algebra/applying-math-reasoning-topic/alternate-number-bases/v/number-systems-introduction

Here is specifically for decimal to binary
(you may need to look at the videos earlier on in the lessons for converting between bases to understand it) :
https://www.khanacademy.org/math/pre-algebra/applying-math-reasoning-topic/alternate-number-bases/v/decimal-to-binary

1 Vote • Comment • Flag 2 months ago by Cameron

Answer this question...