# CodeChef Discussion



## Best known algos for calculating nCr % M

23 29

I have often encountered calculating nCr problems the last one being one of the IIIT-M problems but it probably had weak test cases that got easy solutions AC. The other one https://cs3.interviewstreet.com/challenges/dashboard/#problem/50877a587c389 of codesprint was completey out of my minds and hands , I tried the inverse euler , power of p in n! ( I would also like if someone could explain me the theory of this method ) but all went in vain . Anyone who could suggest probably the best algos for cases when nCr is to be calculated with large n and r with and without % M where M is prime/not prime + the above method which I coded but couldn't understand how it was so.

mathematics runtime algorithm time This question is marked "community wiki".

wikified 18 Nov '12, 09:51

asked 15 Nov '12, 14:45

kavishrox
285●10●17●19
accept rate: 5%



Tags:

algorithm ×923
runtime ×158
time ×115
mathematics ×65

Asked: 15 Nov '12, 14:45
Seen: 30,969 times
Last updated: 11 Jun '15, 12:00

Related questions
how to reduce compilation time in C ??
Generate all the k-element subsets from a n-element set (n>k).
Longest Increasing Subsequence
Is there any algorithm for "Longest increasing subsequence" with complexity less than O(n^2)?
Can anyone suggest me how can i find longest chain of nodes ?
Time Complexity Calculation...
Does the system test solution on all tests?
runtime error in Little Elephant and Divisors of nov cook off?????
Run time error in Fierce Battles
Run time error

### 7 Answers:

oldest newest most voted

96

I encountered $^nC_r$ for the first time on GCJ 08, Round 3 Problem D .. link to analysis.

The first key idea is that of Lucas' Theorem.

Lucas's Theorem reduces $^nC_r$ % M to

$(^{n_0}C_{r_0}$ % M) $(^{n_1}C_{r_1}$ % M) ... $(^{n_k}C_{r_k}$ % M)

Where,

$(n_k n_{k-1} \ldots n_0)$ is the base M representation of n
$(r_k r_{k-1} \ldots r_0)$ is the base M representation of r

- Note, if any of the above terms is zero because $r_i > n_i$ or any other degeneracy, then the binomial coefficient $^nC_r$ % M = 0

This means that any of the terms in the expansion of $^{n_i}C_{r_i}$ is not divisible by M. But this is only half the job done.

Now you have to calculate $^nC_r$ % M (ignoring subscripts for brevity) for some $0 \le r \le n < M$

There are no ways around it, but to calculate

[ n! / ( r! (n-r)! ) ] % M

Without loss of generality, we can assume r ≤ n-r

Remember, you can always do the Obvious. Calculate the binomial and then take a modulo. This is mostly not possible because the binomial will be too large to fit into either int or long long int (and Big Int will be too slow)

This can then be simplified by using some clever ideas from Modular Arithmetic.

- If A*B % M = 1, A and B are modular multiplicative inverses of each other.

For brevity, we say B % M = $A^{-1}$ % M

It is not always possible to calculate modular multiplicative inverses. If A and M are not co-prime, finding a B will not be possible.
For example, A = 2, M = 4. You can never find a number B such that 2*B % 4 = 1

Most problems give us a prime M. This means calculating B is always possible for any A < M.
For other problems, look at the decomposition of M. In the codesprint problem you mentioned

142857 = $3^3$ * 11 * 13 * 37

You can find the result of $^nC_r$ % m for each m = 27, 11, 13, 37. Once you have the answers, you can reconstruct the answer modulo 142857 using Chinese Remainder Theorem. These answers can be found by Naive Methods since, m is small.

I have also seen problems where M is a product of large primes, but square free. In these cases, you can calculate the answers modulo the primes that M is composed of using modular inverses (a little more about that below), and reconstruct the answer using CRT.

I am yet to see a problem where M is neither, but if it is. I do not know if there is a way to calculate binomial coefficients generally (since you cannot calculate modular inverses, and neither can you brote force). I can dream of a problem where there are powers of small primes, but square-free larger ones for a Number Theory extravaganza.

There is one other way to calculate $^nC_r$ for any M which is small enough (say M ≤ 5000) or small n and r (say r ≤ n ≤ 5000) by using the following recursion with memoization

$^{n}C_{r} = {}^{n-1}C_{r} + {}^{n-1}C_{r-1}$

Since there are no divisions involved (no multiplications too) the answer is easy and precise to calculate even if the actual binomials would be very large.

So, back to calculating

[ n! / ( r! (n-r)! ) ] % M, you can convert it to

$n * (n-1) \ldots * (n-r+1) * r^{-1} * (r-1)^{-1} \ldots * 1$

Of course, each product is maintained modulo M.

This may be fast enough for problems where M is large and r is small.

But sometimes, n and r can be very large. Fortunately, such problems always have a small enough M :D

The trick is, you pre-calculate factorials, modulo M and pre-calculate inverse factorials, modulo M.

fact[n] = n * fact[n-1] % M
ifact[n] = modular_inverse(n) * ifact[n-1] % M

Modular Multiplicative Inverse for a prime M is in fact very simple. From Fermat's Little Theorem

$A^{M-1} \% M = 1$

Hence, $A * A^{M-2} \% M = 1$
Or in other words,

$A^{-1} \% M = A^{M-2} \% M$, which is easy (and fast) to find using repeated squaring.

There is one last link I wish to paste to make this complete. Modular inverses can also be found using the Extended Euclid's Algorithm. I have only had to use it once or twice among all the problems I ever solved.

link edited 15 Nov '12, 22:25 answered 15 Nov '12, 18:57

gamabunta ♦♦
2.2k●128●183●169
accept rate: 14%

1 very nicely written @gamabunta. :-)

shivamrana (15 Nov '12, 21:16)

2 @gamabunta: this one is a godd recipe for a tutorial on codechef and topcoder @admins do try to make it a tutorial ..

kavishrox (15 Nov '12, 22:06)

@gamabunta: Firstly Thanks - Well written .

Lucas's Theorem reduces nCr % M to

(n0Cr0 % M) (n1Cr1 % M) ... (nkCrk % M)

Where,

(nknk-1...n0) is the base M representation of n

(rkrk-1...r0) is the base M representation of r

This is only where M is prime .. But in that particular problem M is not prime and so can we reduce it into Base form (And still Lucas Theorem Holds) ? And if we use CRT how can we use it ?

Thanks Anu :)

anudeep2011 (17 Nov '12, 22:49)

@gamabunta: yes probably if you explain chinese remainder that would be even more beneficial...I have read it a lot many times but forget it too easily.

kavishrox (18 Nov '12, 00:13)

1 Amazing Detail. Cheers. Just a great feeling seeing someone spend so much time and energy typing this out for others. :)

P.S. @anudeep2011:Lucas' Theorem holds true even for prime powers(like 27=3^3) (Called generalized Lucas' theorem)

pvaish (14 Mar '14, 01:31)

I agree completely with what @kavishrox wrote...

6

Admins, would it be possible to have some sort of "pin" feature, so that the "tutorial" like posts wouldn't go down? ;)

It would greatly help newbies and ofc make this community even more respected :D

Bruno

link answered 17 Nov '12, 04:56

kuruma
16.8k●72●143●208
accept rate: 8%

To take as example :

1

"142857 = 27 * 11 * 13 * 37. You can find the result of nCr % m for each m = 27, 11, 13, 37."

as 27 is not a prime, i hope we would have to resort to last method of finding all fact[n] and inverse-fact[n] for 27.

So we want nCr and we have x! for all x, WHY do we need inverse factorial ? Is it for (n-r)! and r!. To find inverse of (n-r) and r mod M and then multiply all of their factorials and then find % M.

link edited 18 Nov '12, 15:10 answered 18 Nov '12, 15:08

ashishnegi001
162●1●3●7
accept rate: 0%

what if do not want modulo, just ncr which does't fit in even long long?

1

link

answered 15 Dec '13, 13:31
surajk9035
15●1
accept rate: 0%

Would someone pls xplain me what are these "inverse factorials" ??

0

link

answered 18 Nov '12, 10:10
dg9_17
31●2
accept rate: 0%

How would you calculate nCr mod 27?

0

You can't use the inverse modulo here.

and also one can't do this by nCr=(n-1)Cr+(n-1)C(r-1) given that the minimum space would be just 2 rows but each of size r which in this case is 1000000000?

link

answered 17 Jul '13, 16:53
jaskaran_1
525●23●35●50
accept rate: 0%

What is the fastest among all the ways to calculate the value of nCr mod M? where M is definitely prime?

0

link

answered 04 Apr '15, 06:25
~~imujjwalanand~~
(suspended)
accept rate: 0%

[hide preview]

community wiki

Type the text

Privacy & Terms

reCAPTCHA™

Post Your Answer