L8

abhi shelat

7

7 * 7

$$\underbrace{7 \cdot 7 \cdots 7}_{x \text{ times}}$$

$$7^x$$

7^{1000}


```
125325663996571831810755483238273420616498507508098617146349500752097059631738116432448839054351520763198615919551594076685828989467263022761790838270854579830015111246661203984624358929832571615718014704096305668097507613273663023226895250541385927158426088684494082416768617708189592286936039922311125683719215046689156738352590137241554510185855964549927575493247391132548534378497978806084951085874202011836362315727420109554782988791530088289711844550500230485638413189947132142243947334199259300735622492937419453650061490302105127920314430401636855677549136337481321811349678427076091437345045399337348611261168055929355402992823192491190360027036112283180935872775214517464013178274657100736321564606838252739601156414628445543663144696050650160812621814327062666195172701780200286645023823083185928061371310300829284071141207731280600001
```


7^{10000}


```
955872885674440970277216219337772744771565088008376702328097477368828026889847379693518728469156781763033915470324183094405583670617788098960565377878039968120853046553689992387208372835669135037243033769661992663921364613620578078079956465825388253571736411644948206142873619873902998932401372061622798495221711362842179827842727414295003904969814698887647607489396731788861520661760706501004653662221061702731812881848142381806779161139684997815338572295836554681424837278459728932615631869144483961065493749103944084960759707289272096063771350099293437708895448863952101969852485119002913967233458109697183426176100815914953448411817773659723619107102278444558710596976906677394315185249714607867087670127235754713322083182486995225340569547224637119563739118489076548828690529780180285781790200709105216646950073585170470697772749289033207716010189419921643140458588853251039197496070544889716525591702409353851295924370666649874439353143103187877095721819476871104603962154142099655233744745835520269309421928095461368238705472240356556168312930388055119358800784517897136666191879774303837940606358829794421599264636917853406249677434941352456006868354469137172102765364263540581034439147785528571831089732436874817770373138329060664619624176632227394978942718649457391622038713205017081287426068599398078599136506248490477428409273845322293948595250418133360588306541097125781628428197968967708785261938674580158414216117184233630606194460758031603508715854504925822065735415717117642951641517044516126295875685241125426213156539867646429188235196051228173319647034882976905185935972581124925543404862657129745082863682768217744513289968936319344348417912106139777376465001699264700607259391929003413720980579337938903285241069450117215301586795975493669615443235203961082168458566029344221633225550705805007157057934008407572852685095108138875038783467527808394272833031297114865113153007797888201808406197212237152401786723674966370422728602922450067577330084637312473407306058275594314586875447869409332649806214474569067079915471288310333702938311987897305665292222672504649400164437794374345278353386665833282569911552610248180874227131635408318821721673507240125507583163850231515419634983097154734421315315036556738503952579851113060316168285064441232537103590182621079377832843019727603226096866723988722891001182616180230977331759604555319089344641568797218346060486917623879348939899420832926786114165436117688507634248134379080283752266998295759671652312537861057724541288910048170317202109146415483638475418330792268525776550860856856297578230069719720403254069859443028697721989276237729544128518104287912780299902471908061321027142210255474075271028994680673537259392939342492022814681786038589505372793746046900629086230630259659924116133599305462808406201633629428637101656936527502479985898768332184004335966399236438077935374851225863790788329952053631976486273900477504002455220369355782732974197522274191043597836052137499425803610123172609307225829744073565768765665341704052432631923216651084394928694118353258945746614934400239854204503176898282013448080753378830552236604912047688409349395104864407345150328046921038566652090310866567690216457225948387309395598104621188922299520026018146333402349443994282604600758932366539153393090504007103991188634801741357291263542723275261989318470204566728795325066963336117780662108205792489168581971029055585610731736194240588491958013125492135128060000001
```

WolframAlpha computational knowledge engine

7^100000000000000000

Examples Random

Input:

$7^{100\,000\,000\,000\,000\,000}$

Power of 10 representation:

$10^{10^{10^{1.2285776105298 23}}}$

Number length:

84 509 804 001 425 684 decimal digits

$\approx 8.45098 \times 10^{16}$ digits

Last few digits:

...0000000001

Computed by Wolfram *Mathematica*

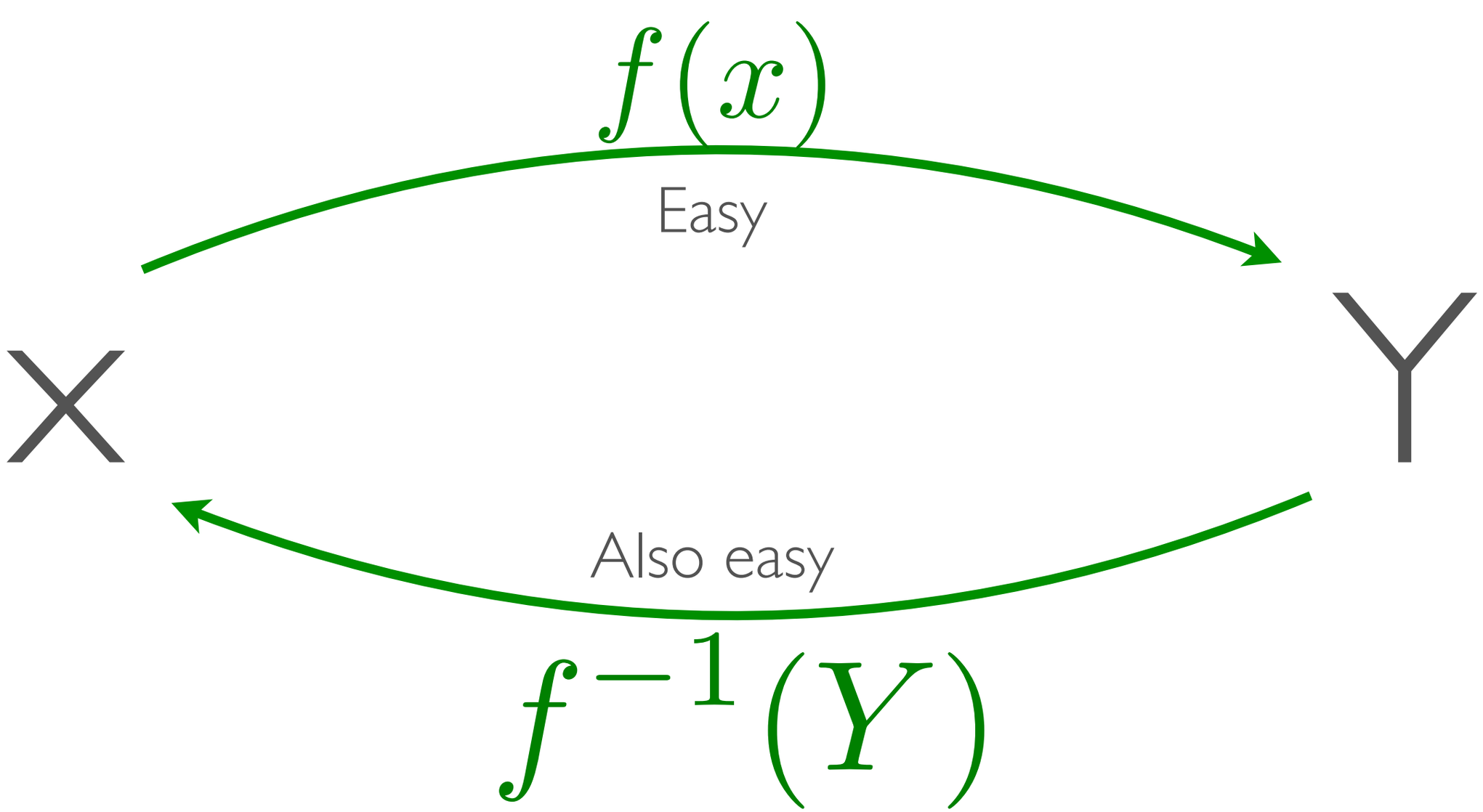
Download page

Inverse problem is EASY:

Given: 1231928

Find X such that $7^X = 1231928$

7.20695710776661638632370146569432966292556545698087919670 51



$$7^x$$

$$\underbrace{7 \cdot 7 \cdots 7}_{x \text{ times}}$$

$$7^x \bmod p$$

$$7^2 = 7 \cdot 7 \mod 131$$

$$7^2 = 7 \cdot 7 \mod 131$$

$$= 49$$

$$7^3 = 49 \cdot 7 \bmod 131$$

$$343 \bmod 131$$

$$7^3 = 49 \cdot 7 \bmod 131$$

$$\begin{array}{r|l} & 2 \\ \hline 131 & 343 \\ & 262 \\ \hline & 81 \end{array} \qquad \bmod 131$$

$$7^3 = 49 \cdot 7 \mod 131$$

$$\begin{array}{r|l} & 2 \\ \hline 131 & 343 \\ & 262 \\ \hline & 81 \end{array} \mod 131$$

$$= 81 \mod 131$$

Pick a large prime number, ~200 digits long

p=
```
52061717268811022582985085340962890043057948799610254
54308897584403272194404580619934826278347900750288237
80618996828085538214474401045588718257206979528176365
69237467061923678480735550795549941664404569955484 99
```

$$(7, x, p) \rightarrow 7^x \bmod p$$

Remarkably easy

Assailing U.S. and Kiev, Putin Keeps Open Option of Force, By STEVEN LEE MYERS, ELLEN BARRY and ALAN COWELL 56 minutes ago, President Vladimir V. Putin of Russia on Tuesday described events in Ukraine as an unconstitutional coup, expressed contempt toward the United States and said that any potential use of military force would be a last resort.

```
sage:
power_mod(7,66161598060906110332854683463298110032760602184432811817061032760202131532801302103280131706121033120232711215000144326721328384698669783276696932778969828344326976766978326665828289329811003265766578326779876976763253543310061118170215329804114432811502160601021116328708980106100614328646328118170610331202328318161605973312103285180216009821330102160015059902003302190211171533061032860814980611013298153298103318110012111617061718170612109808330012180124433022113150216160200330012111702101316331712199815003317050132861106170200328416981702153298110033159806003317049816329811213313121702111705980833181601331202331006090616981521330312150001332012180900329901329733089816163315021612151646,p)

4979297116858421752786333230062932909865498326942335772959528902620394112679338715866175118050970123525663342996263555367600208386959076972752226702772583539808922335684593676931602507077210362685189934639324 29
```

$Y = 12345$

$$(7, p, Y) \longrightarrow x$$

such that $7^x \bmod p = Y$

Incredibly hard!

World record in discrete logarithms in GF(p)

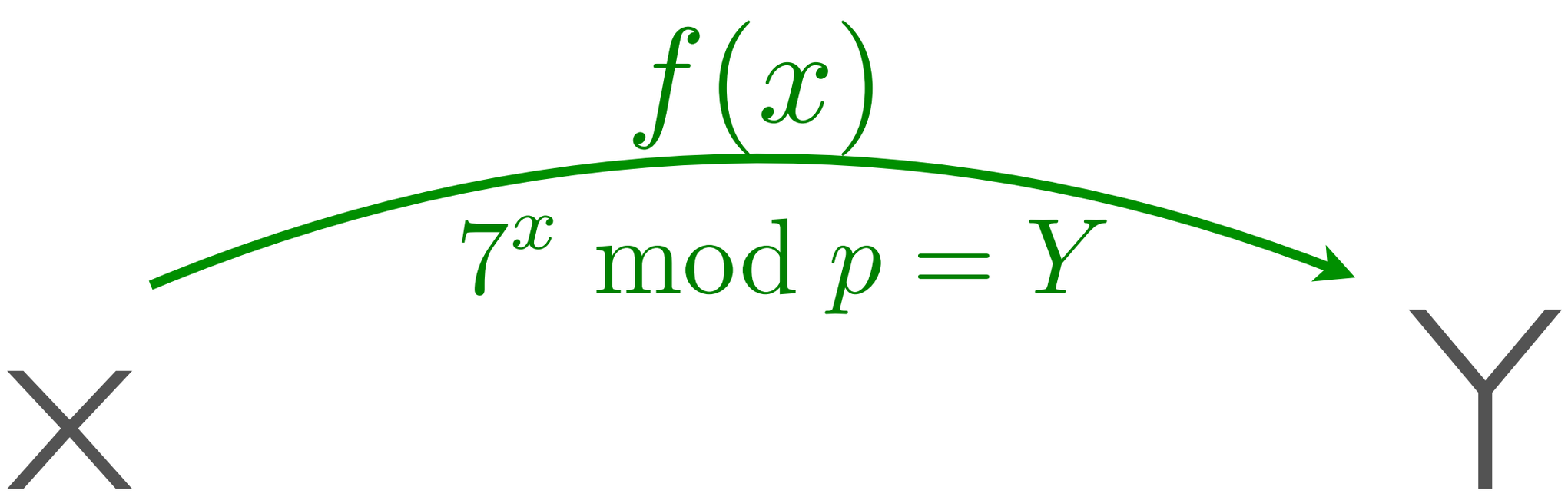
232 digits (768 bits)

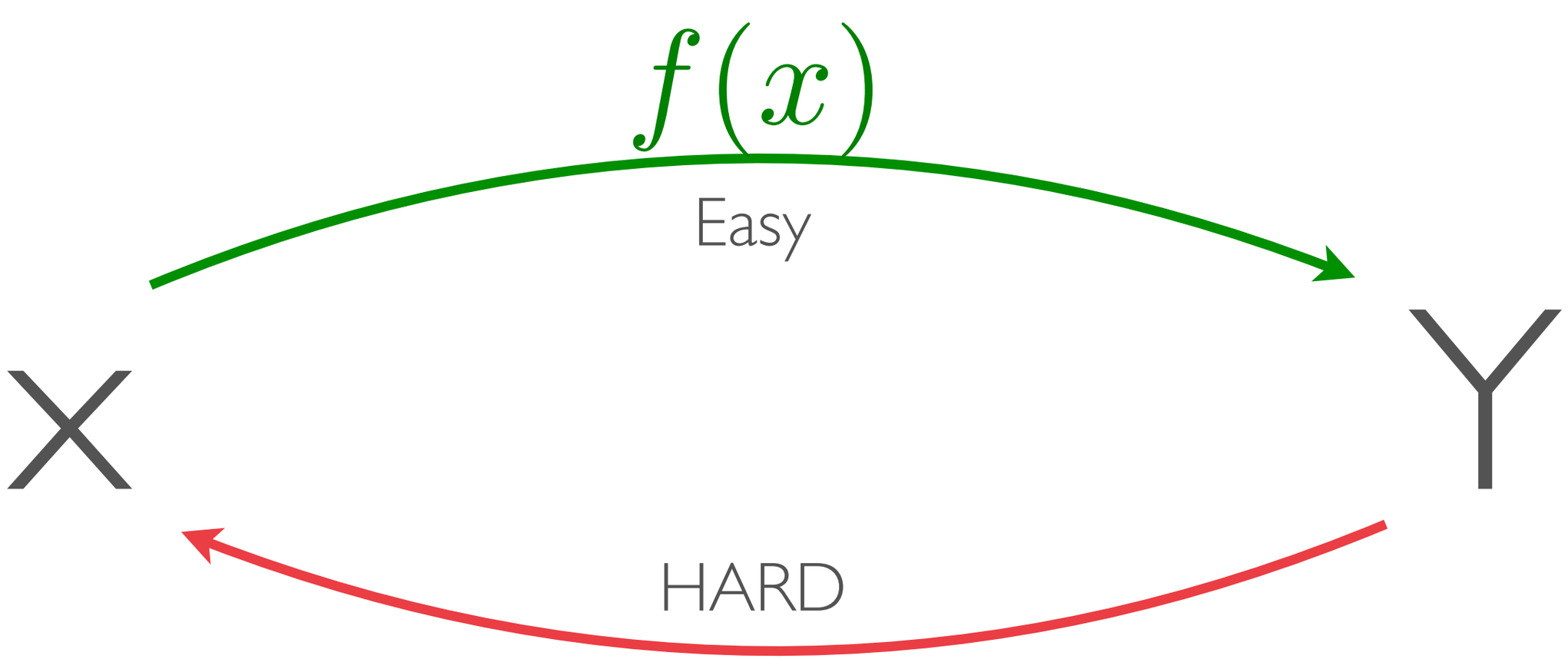
6600yrs of CPU time

Intel Xeon E5-2660 at 2.2 GHz

2016

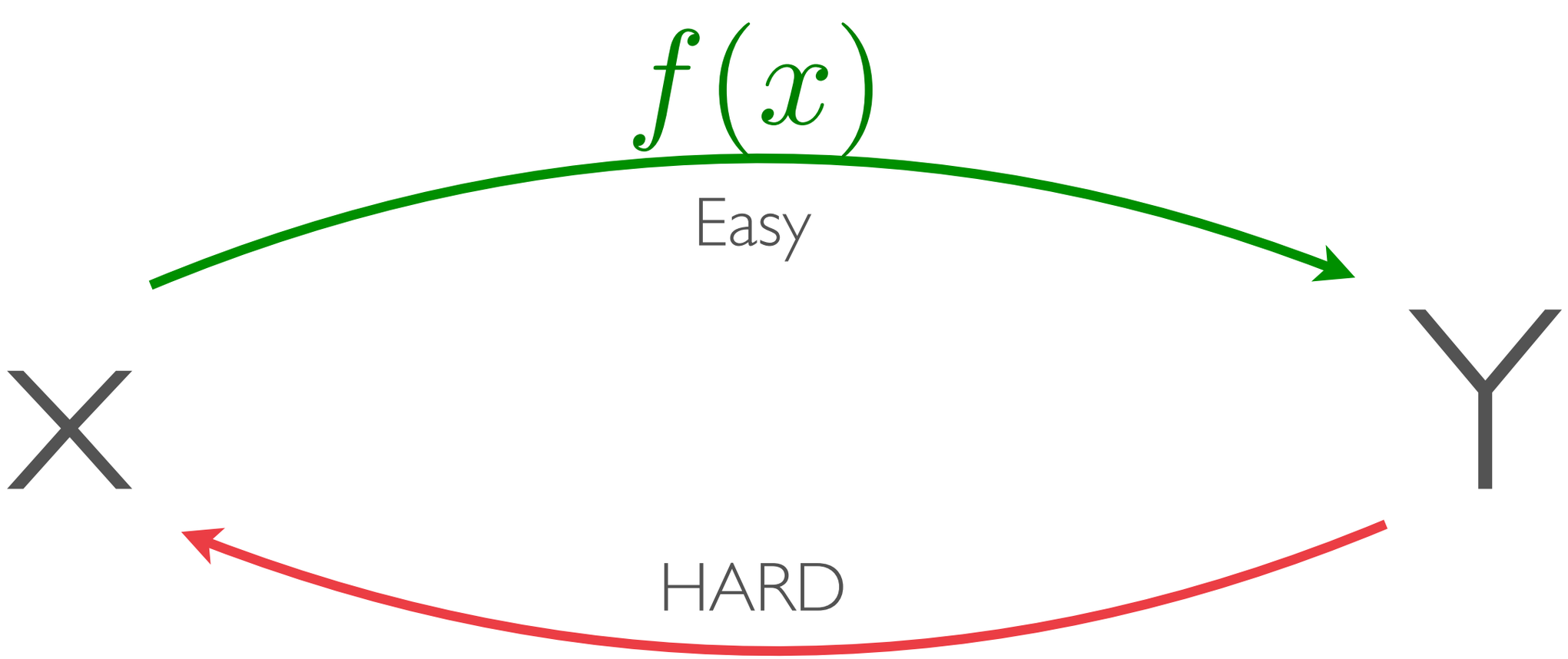
Thorsten Kleinjung, Claus Diem, Arjen K. Lenstra, Christine Priplata, and Colin Stahlke

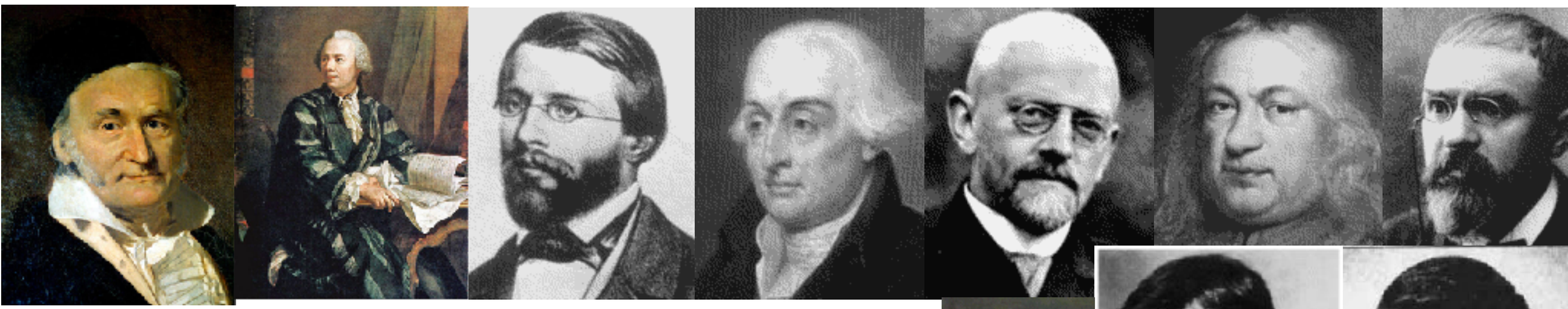

$f(x)$
$7^x \bmod p = Y$
X
Y



compute an x such that $7^x = Y \bmod p$

http://www.fitzmuseum.cam.ac.uk/gallery/chinesevases/sortingfragments.html





http://fabpedigree.com/james/mathmen.htm

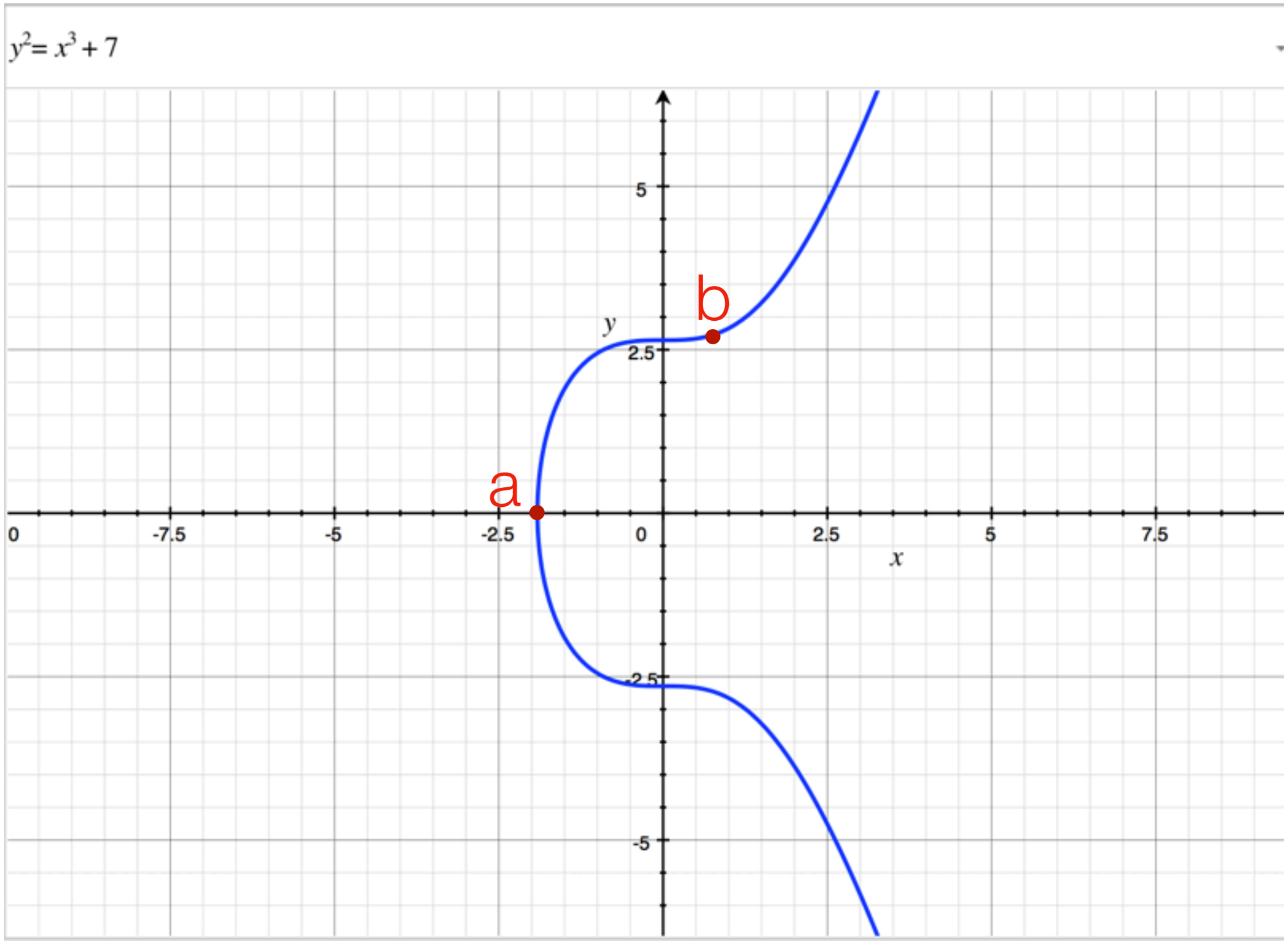
http://upload.wikimedia.org/wikipedia/commons/thumb/9/9b/Carl_Friedrich_Gauss.jpg/440px-Carl_Friedrich_Gauss.jpg

Same problem, harder
(seemingly)

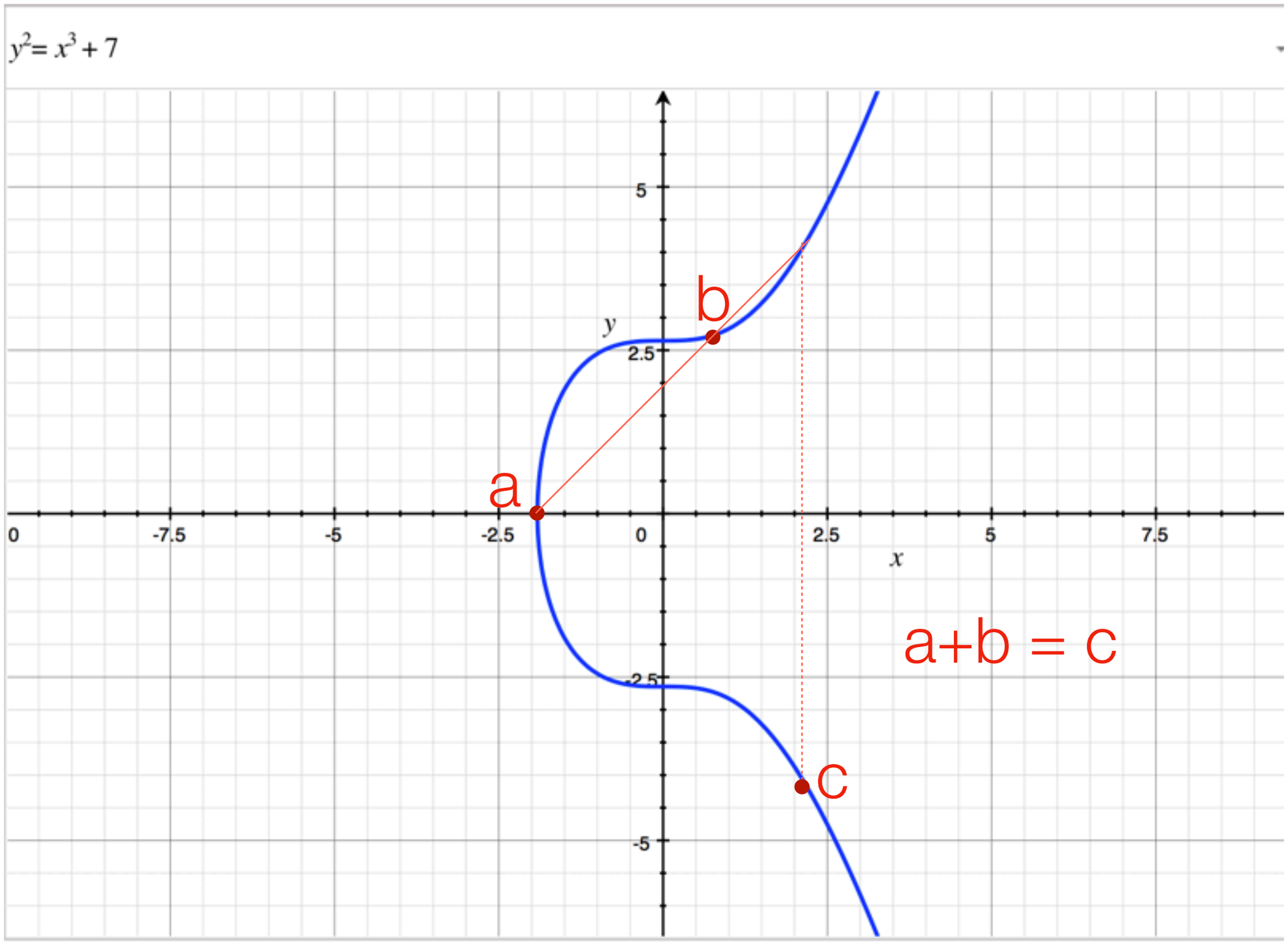
Instead of using 7 (integers mod p), use (x,y) points that lie on an elliptic curve such as

$$y^2 = x^3 + 7$$

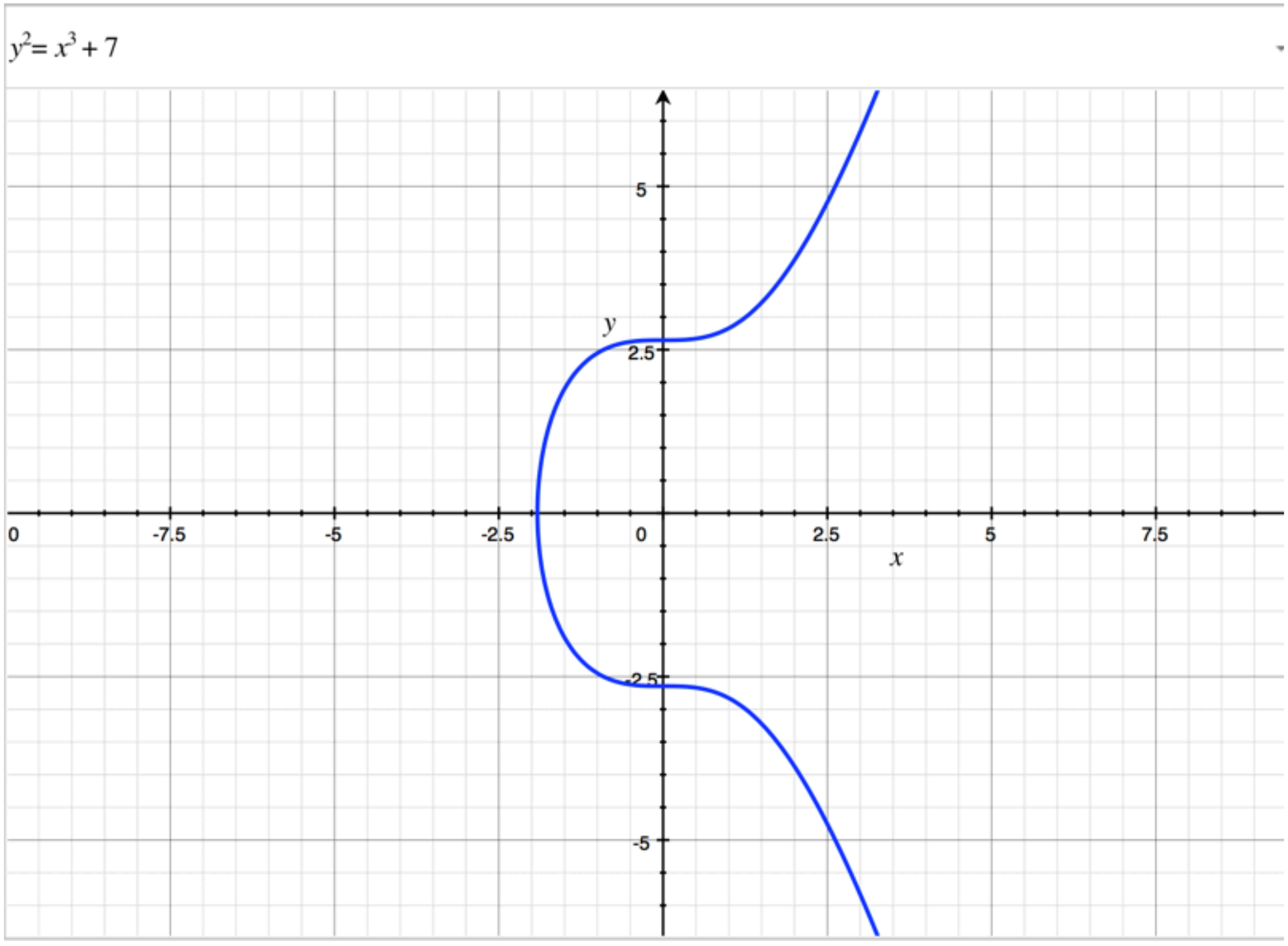
Points on such curves can be added



Points on such curves can be added

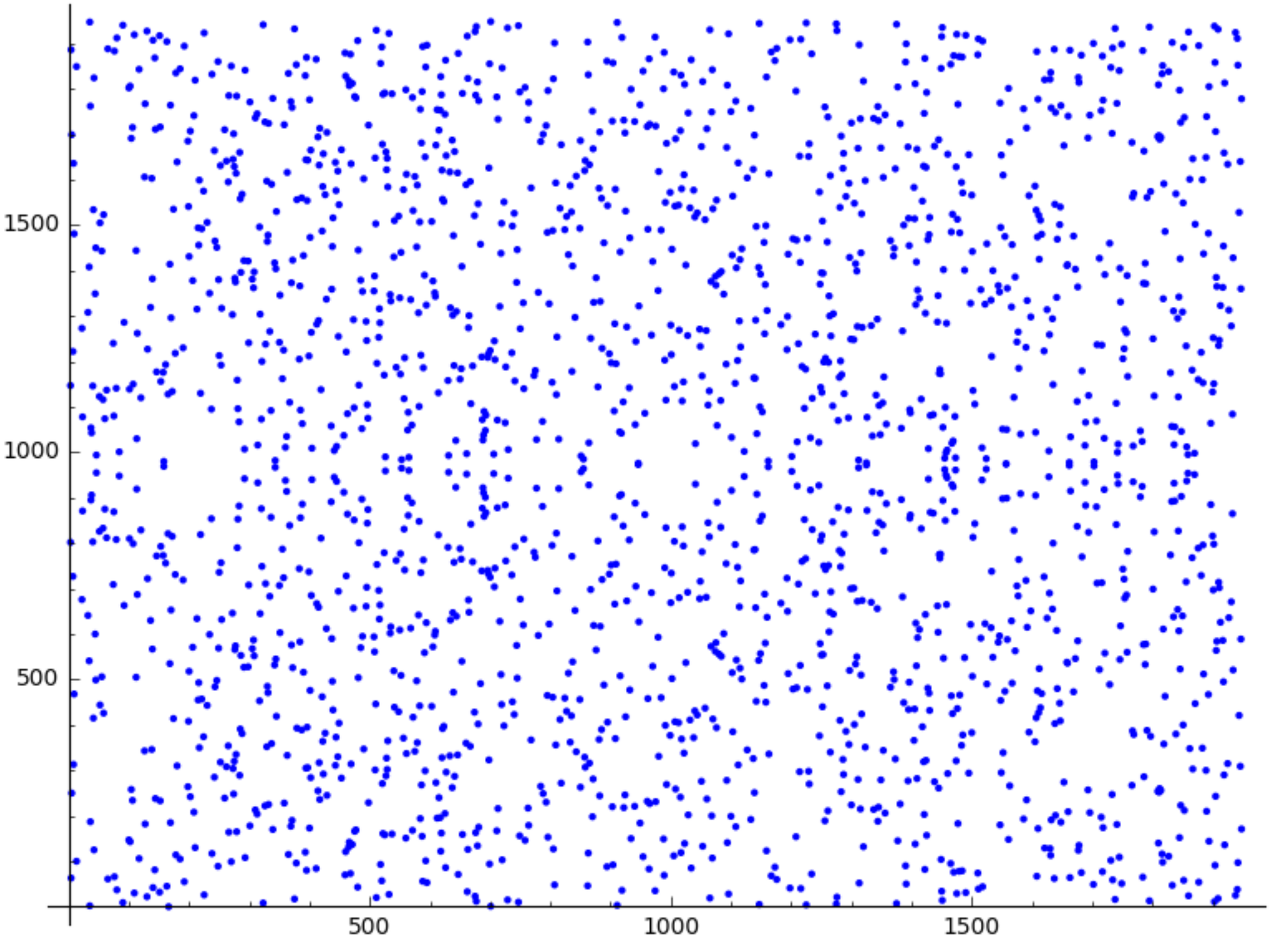


Problem: infinite precision math



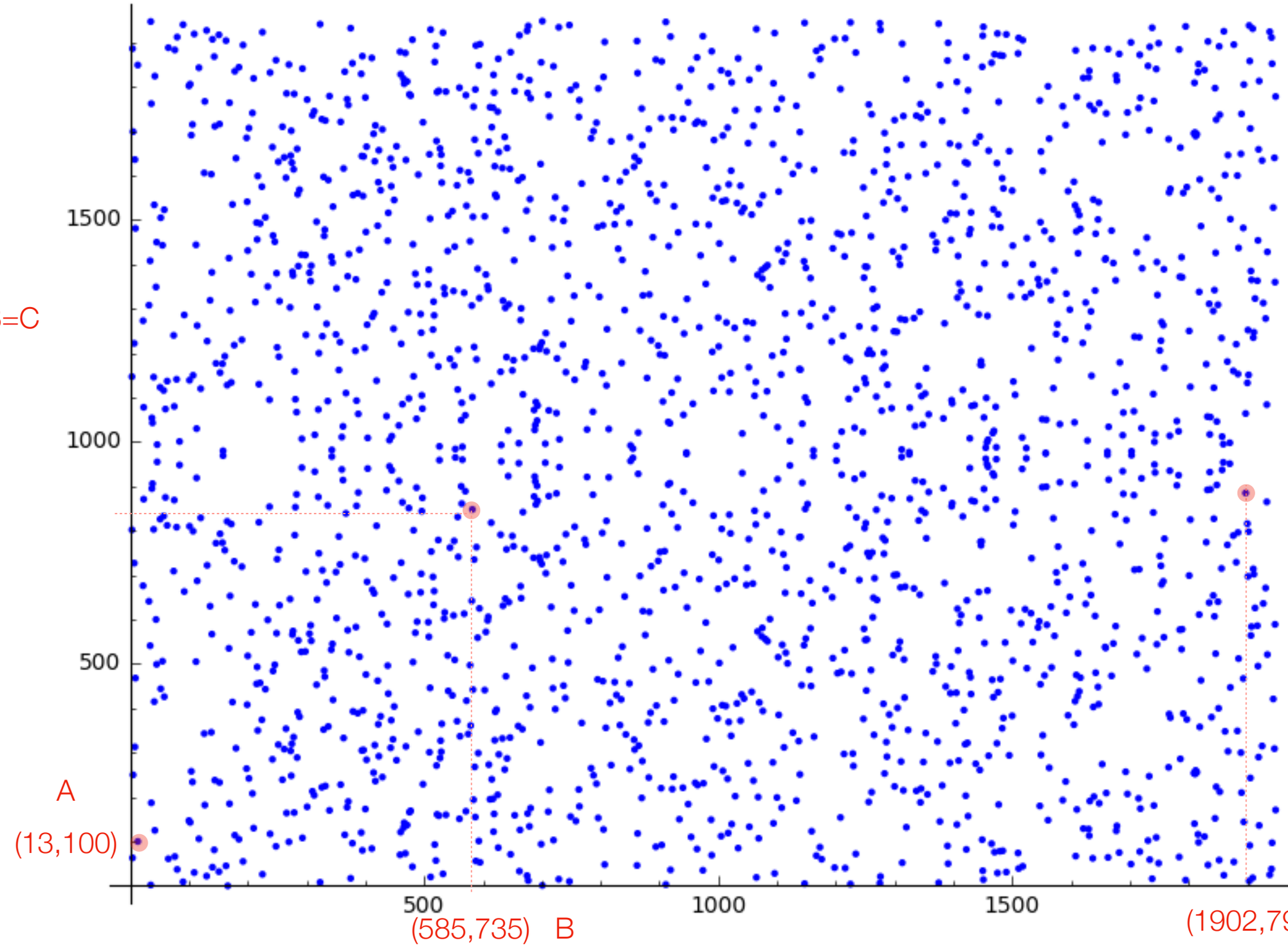
Solution: mod p

$$y^2 = x^3 + 7 \bmod 1949$$



Solution: mod p

$$y^2 = x^3 + 7 \bmod 1949$$



A+B=C

Use a big prime

$$y^2 = x^3 + 7 \mathrm{mod}$$

115,792,089,237,316,195,42
3,570,985,008,687,907,853,
269,984,665,640,564,039,45
7,584,007,908,834,671,663

$$2^{256} - 2^{32} - 2^9 - 2^8 - 2^7 - 2^6 - 2^4 - 1$$

The hard problem

Given the points A and Y = $\underbrace{A+A+A+\ldots+A}_{x \text{ times}}$

Find x.

This is the discrete log problem on elliptic curves

ECDSA signature scheme

We can use this hard problem to make a signature scheme.

One standard way to do this is ECDSA.

This is the sig scheme used in Bitcoin, Ethereum, Litecoin, etc.

ECDSA Gen

Recall Gen() produces (sk, pk).

Let G be the elliptic curve and g a generator of G.

Gen(): Pick a random x in [1..q]

ECDSA Gen

Recall Gen() produces (sk, pk).

Let G be the elliptic curve and g a generator of G.

Gen(): Pick a random x in [1..q]

What is q? It is the *order* of g, a number very close to p.

ECDSA Gen

Recall Gen() produces (sk, pk).

Let G be the elliptic curve and g a generator of G.

Gen(): Pick a random x in [1..q]

What is q? It is the *order* of g, a number very close to p.

Output pk= x*g. [that is, g+g+g+…+g (x times)]

ECDSA Gen

Recall Gen() produces (sk, pk).

Let G be the elliptic curve and g a generator of G.

Gen(): Pick a random x in [1..q]

What is q? It is the *order* of g, a number very close to p.

Output pk= x*g. [that is, g+g+g+…+g (x times)]

Output sk= x

ECDSA sign

These steps seem magical until you understand how and why they work (in a crypto class).

Sign(sk, msg): z = hash(msg)
k = pick random # in [1..q-1]
r = “x” coordinate of k*g,
$s = k^{-1}(z + r*sk) \bmod q$
Output (r,s) as signature.

If the EC DLOG problem is hard, and hash is secure, this scheme is shown to be secure against forgeries.


abhi — -bash — 125×27
ab2017:abhi abhi$


abhi — -bash — 125×27
ab2017:abhi abhi$