MAS114: Lecture 18

James Cranch

http://cranch.staff.shef.ac.uk/mas114/

2021–2022

Later this week

Later this week

I will be on strike 1st—3rd December, as part of UCU's industrial dispute over pay, pensions and working conditions.

Later this week

I will be on strike 1st—3rd December, as part of UCU's industrial dispute over pay, pensions and working conditions.
I will not be working (or getting paid) on those days.

Squaring mod p

Squaring mod p

We worked with the factorial in the proof of Fermat's Little Theorem without ever needing to calculate it.

Squaring mod p

We worked with the factorial in the proof of Fermat's Little Theorem without ever needing to calculate it. It turns out we *can* calculate it, using a clever trick.

Squaring mod p

We worked with the factorial in the proof of Fermat's Little Theorem without ever needing to calculate it. It turns out we *can* calculate it, using a clever trick.
However, we'll need a fact first:

Squaring mod p

We worked with the factorial in the proof of Fermat's Little Theorem without ever needing to calculate it. It turns out we *can* calculate it, using a clever trick.
However, we'll need a fact first:

Proposition

Let p be a prime, and let a be an integer with the property that $a^2 \equiv 1 \pmod{p}$. Then either $a \equiv 1 \pmod{p}$ or $a \equiv -1 \pmod{p}$.

Squaring mod p

We worked with the factorial in the proof of Fermat's Little Theorem without ever needing to calculate it. It turns out we *can* calculate it, using a clever trick.
However, we'll need a fact first:

Proposition

Let p be a prime, and let a be an integer with the property that $a^2 \equiv 1 \pmod{p}$. Then either $a \equiv 1 \pmod{p}$ or $a \equiv -1 \pmod{p}$.

Proof.

If $a^2 \equiv 1 \pmod{p}$, then $a^2 - 1 \equiv 0 \pmod{p}$

Squaring mod p

We worked with the factorial in the proof of Fermat's Little Theorem without ever needing to calculate it. It turns out we *can* calculate it, using a clever trick.
However, we'll need a fact first:

Proposition

Let p be a prime, and let a be an integer with the property that $a^2 \equiv 1 \pmod{p}$. Then either $a \equiv 1 \pmod{p}$ or $a \equiv -1 \pmod{p}$.

Proof.

If $a^2 \equiv 1 \pmod{p}$, then $a^2 - 1 \equiv 0 \pmod{p}$, ie
$(a-1)(a+1) \equiv 0 \pmod{p}$.

Squaring mod p

We worked with the factorial in the proof of Fermat's Little Theorem without ever needing to calculate it. It turns out we *can* calculate it, using a clever trick.
However, we'll need a fact first:

Proposition

Let p be a prime, and let a be an integer with the property that $a^2 \equiv 1 \pmod{p}$. Then either $a \equiv 1 \pmod{p}$ or $a \equiv -1 \pmod{p}$.

Proof.

If $a^2 \equiv 1 \pmod{p}$, then $a^2 - 1 \equiv 0 \pmod{p}$, ie $(a-1)(a+1) \equiv 0 \pmod{p}$. In other words, $p \mid (a-1)(a+1)$.

Squaring mod p

We worked with the factorial in the proof of Fermat's Little Theorem without ever needing to calculate it. It turns out we *can* calculate it, using a clever trick.
However, we'll need a fact first:

Proposition

Let p be a prime, and let a be an integer with the property that $a^2 \equiv 1 \pmod{p}$. *Then either* $a \equiv 1 \pmod{p}$ *or* $a \equiv -1 \pmod{p}$.

Proof.

If $a^2 \equiv 1 \pmod{p}$, then $a^2 - 1 \equiv 0 \pmod{p}$, ie $(a-1)(a+1) \equiv 0 \pmod{p}$. In other words, $p \mid (a-1)(a+1)$.
But then, either $p \mid a - 1$

Squaring mod p

We worked with the factorial in the proof of Fermat's Little Theorem without ever needing to calculate it. It turns out we *can* calculate it, using a clever trick.
However, we'll need a fact first:

Proposition

Let p be a prime, and let a be an integer with the property that $a^2 \equiv 1 \pmod{p}$. Then either $a \equiv 1 \pmod{p}$ or $a \equiv -1 \pmod{p}$.

Proof.

If $a^2 \equiv 1 \pmod{p}$, then $a^2 - 1 \equiv 0 \pmod{p}$, ie $(a-1)(a+1) \equiv 0 \pmod{p}$. In other words, $p \mid (a-1)(a+1)$.
But then, either $p \mid a - 1$ (in which case $a \equiv 1 \pmod{p}$)

Squaring mod p

We worked with the factorial in the proof of Fermat's Little Theorem without ever needing to calculate it. It turns out we *can* calculate it, using a clever trick.
However, we'll need a fact first:

Proposition

Let p be a prime, and let a be an integer with the property that $a^2 \equiv 1 \pmod{p}$. Then either $a \equiv 1 \pmod{p}$ or $a \equiv -1 \pmod{p}$.

Proof.

If $a^2 \equiv 1 \pmod{p}$, then $a^2 - 1 \equiv 0 \pmod{p}$, ie
$(a-1)(a+1) \equiv 0 \pmod{p}$. In other words, $p \mid (a-1)(a+1)$.
But then, either $p \mid a - 1$ (in which case $a \equiv 1 \pmod{p}$), or
$p \mid a + 1$

Squaring mod p

We worked with the factorial in the proof of Fermat's Little Theorem without ever needing to calculate it. It turns out we *can* calculate it, using a clever trick.
However, we'll need a fact first:

Proposition

Let p be a prime, and let a be an integer with the property that $a^2 \equiv 1 \pmod{p}$. Then either $a \equiv 1 \pmod{p}$ or $a \equiv -1 \pmod{p}$.

Proof.

If $a^2 \equiv 1 \pmod{p}$, then $a^2 - 1 \equiv 0 \pmod{p}$, ie $(a-1)(a+1) \equiv 0 \pmod{p}$. In other words, $p \mid (a-1)(a+1)$. But then, either $p \mid a-1$ (in which case $a \equiv 1 \pmod{p}$), or $p \mid a+1$ (in which case $a \equiv -1 \pmod{p}$). □

Some comments

Some comments

Remark

This theorem is not true for some composite moduli!

Some comments

Remark

This theorem is not true for some composite moduli! For example, $1^2 \equiv 3^2 \equiv 5^2 \equiv 7^2 \equiv 1 \pmod 8$.

Some comments

Remark

This theorem is not true for some composite moduli! For example, $1^2 \equiv 3^2 \equiv 5^2 \equiv 7^2 \equiv 1 \pmod{8}$.

I regard this as more evidence that prime moduli behave very nicely indeed!

Some comments

Remark

This theorem is not true for some composite moduli! For example, $1^2 \equiv 3^2 \equiv 5^2 \equiv 7^2 \equiv 1 \pmod 8$.
I regard this as more evidence that prime moduli behave very nicely indeed!

Remark

This means that if we have a not congruent to ± 1 modulo a prime p, then the inverse of a (modulo p) is different to a.

Some comments

Remark

This theorem is not true for some composite moduli! For example, $1^2 \equiv 3^2 \equiv 5^2 \equiv 7^2 \equiv 1 \pmod{8}$.

I regard this as more evidence that prime moduli behave very nicely indeed!

Remark

This means that if we have a not congruent to ± 1 modulo a prime p, then the inverse of a (modulo p) is different to a.

Indeed, if $a \equiv a^{-1}$ then $1 \equiv aa^{-1} \equiv a^2$.

Some comments

Remark

This theorem is not true for some composite moduli! For example, $1^2 \equiv 3^2 \equiv 5^2 \equiv 7^2 \equiv 1 \pmod 8$.
I regard this as more evidence that prime moduli behave very nicely indeed!

Remark

This means that if we have a not congruent to ± 1 modulo a prime p, then the inverse of a (modulo p) is different to a.
Indeed, if $a \equiv a^{-1}$ then $1 \equiv aa^{-1} \equiv a^2$.

Now, this allows us to do this:

Wilson's Theorem

Wilson's Theorem

Theorem (Wilson's Theorem)

We have $(n-1)! \equiv -1 \pmod{n}$ *if and only if n is prime.*

Wilson's Theorem

Theorem (Wilson's Theorem)

We have $(n-1)! \equiv -1 \pmod{n}$ *if and only if n is prime.*

Proof.

I'll show firstly that if n is composite, we don't get $(n-1)! \equiv -1 \pmod{n}$.

Wilson's Theorem

Theorem (Wilson's Theorem)

We have $(n-1)! \equiv -1 \pmod{n}$ *if and only if n is prime.*

Proof.

I'll show firstly that if n is composite, we don't get $(n-1)! \equiv -1 \pmod{n}$.

Indeed, suppose that n has a factor a such that $1 < a < n$. Then we certainly have $a \mid (n-1)!$, and so $(n-1)! \equiv 0 \pmod{a}$.

Wilson's Theorem

Theorem (Wilson's Theorem)

We have $(n-1)! \equiv -1 \pmod{n}$ *if and only if n is prime.*

Proof.

I'll show firstly that if n is composite, we don't get $(n-1)! \equiv -1 \pmod{n}$.

Indeed, suppose that n has a factor a such that $1 < a < n$. Then we certainly have $a \mid (n-1)!$, and so $(n-1)! \equiv 0 \pmod{a}$.

However, if $(n-1)! \equiv -1 \pmod{n}$ and $a|n$, then $(n-1)! \equiv -1 \pmod{a}$, which gives a contradiction.

Wilson's Theorem

Theorem (Wilson's Theorem)

We have $(n-1)! \equiv -1 \pmod{n}$ *if and only if* n *is prime.*

Proof.

Wilson's Theorem

Theorem (Wilson's Theorem)

We have $(n-1)! \equiv -1 \pmod{n}$ *if and only if n is prime.*

Proof.

Now I'll show that if n is prime we do get $(n-1)! \equiv -1 \pmod{n}$.

Wilson's Theorem

Theorem (Wilson's Theorem)

We have $(n-1)! \equiv -1 \pmod{n}$ *if and only if n is prime.*

Proof.

Now I'll show that if n is prime we do get $(n-1)! \equiv -1 \pmod{n}$.
Given that n is prime, the product

$$(n-1)! = 1 \cdot 2 \cdot \cdots \cdot (n-1)$$

consists of one representative of each invertible residue class.

Wilson's Theorem

Theorem (Wilson's Theorem)

We have $(n-1)! \equiv -1 \pmod{n}$ *if and only if n is prime.*

Proof.

Now I'll show that if n is prime we do get $(n-1)! \equiv -1 \pmod{n}$. Given that n is prime, the product

$$(n-1)! = 1 \cdot 2 \cdot \cdots \cdot (n-1)$$

consists of one representative of each invertible residue class. We can pair each up with its inverse; each element gets paired with another, except for 1 and -1.

Wilson's Theorem

Theorem (Wilson's Theorem)

We have $(n-1)! \equiv -1 \pmod{n}$ *if and only if n is prime.*

Proof.

Now I'll show that if n is prime we do get $(n-1)! \equiv -1 \pmod{n}$. Given that n is prime, the product

$$(n-1)! = 1 \cdot 2 \cdot \dots \cdot (n-1)$$

consists of one representative of each invertible residue class. We can pair each up with its inverse; each element gets paired with another, except for 1 and -1. So, the product consists of a lot of pairs of inverses (whose product modulo n is 1), together with the odd ones out 1 and -1: so the product is -1 as claimed. □

Examples and remarks

Examples and remarks

Here are some examples:

Examples and remarks

Here are some examples:

- $9! = 362880 \equiv 0 \pmod{10}$, and so 10 is composite.

Examples and remarks

Here are some examples:

- $9! = 362880 \equiv 0 \pmod{10}$, and so 10 is composite.
- $10! = 3628800 \equiv -1 \pmod{11}$, and so 11 is prime.

Examples and remarks

Here are some examples:

- $9! = 362880 \equiv 0 \pmod{10}$, and so 10 is composite.
- $10! = 3628800 \equiv -1 \pmod{11}$, and so 11 is prime. Indeed, 2 and 6 are inverses,

Examples and remarks

Here are some examples:

- $9! = 362880 \equiv 0 \pmod{10}$, and so 10 is composite.
- $10! = 3628800 \equiv -1 \pmod{11}$, and so 11 is prime. Indeed, 2 and 6 are inverses, and 3 and 4,

Examples and remarks

Here are some examples:

- $9! = 362880 \equiv 0 \pmod{10}$, and so 10 is composite.
- $10! = 3628800 \equiv -1 \pmod{11}$, and so 11 is prime. Indeed, 2 and 6 are inverses, and 3 and 4, and 5 and 9,

Examples and remarks

Here are some examples:

- $9! = 362880 \equiv 0 \pmod{10}$, and so 10 is composite.
- $10! = 3628800 \equiv -1 \pmod{11}$, and so 11 is prime. Indeed, 2 and 6 are inverses, and 3 and 4, and 5 and 9, and 7 and 8.

Examples and remarks

Here are some examples:

- $9! = 362880 \equiv 0 \pmod{10}$, and so 10 is composite.
- $10! = 3628800 \equiv -1 \pmod{11}$, and so 11 is prime. Indeed, 2 and 6 are inverses, and 3 and 4, and 5 and 9, and 7 and 8.

Remark

You could use this as a way of testing if a number is prime.

Examples and remarks

Here are some examples:

- $9! = 362880 \equiv 0 \pmod{10}$, and so 10 is composite.
- $10! = 3628800 \equiv -1 \pmod{11}$, and so 11 is prime. Indeed, 2 and 6 are inverses, and 3 and 4, and 5 and 9, and 7 and 8.

Remark

You could use this as a way of testing if a number is prime.
As a matter of fact, it's not a good way of doing it: if we want to check a large number N, it's quicker to do trial division to see if N has any factors, than it is to multiply lots of numbers together.

Examples and remarks

Here are some examples:

- $9! = 362880 \equiv 0 \pmod{10}$, and so 10 is composite.
- $10! = 3628800 \equiv -1 \pmod{11}$, and so 11 is prime. Indeed, 2 and 6 are inverses, and 3 and 4, and 5 and 9, and 7 and 8.

Remark

You could use this as a way of testing if a number is prime.
As a matter of fact, it's not a good way of doing it: if we want to check a large number N, it's quicker to do trial division to see if N has any factors, than it is to multiply lots of numbers together.
But this result was psychologically important in the development of modern fast primality tests: it was the first evidence that there are ways of investigating whether a number N is prime or not by looking at how arithmetic modulo N behaves.

Cryptography

Cryptography

In this section, we'll show off a major modern application of all the ideas above.

Cryptography

In this section, we'll show off a major modern application of all the ideas above. The aim is to talk about (one small but key part of) modern cryptography.

Cryptography

In this section, we'll show off a major modern application of all the ideas above. The aim is to talk about (one small but key part of) modern cryptography.

Cryptography is the study of how to send messages in a form which cannot be read except by the intended recipients.

Cryptography

In this section, we'll show off a major modern application of all the ideas above. The aim is to talk about (one small but key part of) modern cryptography.

Cryptography is the study of how to send messages in a form which cannot be read except by the intended recipients. To *encrypt* the messages is to put them in a form which cannot be read easily;

Cryptography

In this section, we'll show off a major modern application of all the ideas above. The aim is to talk about (one small but key part of) modern cryptography.

Cryptography is the study of how to send messages in a form which cannot be read except by the intended recipients. To *encrypt* the messages is to put them in a form which cannot be read easily; to *decrypt* the messages is to take such messages and recover them in readable form.

Dramatis personae

Dramatis personae

The literature of cryptography usually talks about three people:

Dramatis personae

The literature of cryptography usually talks about three people:

- **Alice** who wishes to send a private message to Bob,

Dramatis personae

The literature of cryptography usually talks about three people:

- **Alice** who wishes to send a private message to Bob,
- **Bob** who wishes to receive a private message from Alice, and

Dramatis personae

The literature of cryptography usually talks about three people:

- **Alice** who wishes to send a private message to Bob,
- **Bob** who wishes to receive a private message from Alice, and
- **Eve** who wishes to find out what Alice is telling Bob.

Dramatis personae

The literature of cryptography usually talks about three people:

- **Alice** who wishes to send a private message to Bob,
- **Bob** who wishes to receive a private message from Alice, and
- **Eve** who wishes to find out what Alice is telling Bob.

Alice and Bob are of course named so that the message goes from A to B.

Dramatis personae

The literature of cryptography usually talks about three people:

- **Alice** who wishes to send a private message to Bob,
- **Bob** who wishes to receive a private message from Alice, and
- **Eve** who wishes to find out what Alice is telling Bob.

Alice and Bob are of course named so that the message goes from A to B. Eve is so named because she is an *eavesdropper*, or perhaps because she is *evil*.

Old-time cryptography

Old-time cryptography

In the olden days, Alice and Bob would have come up with some kind of system depending on a shared secret *key* with which you could encrypt and decrypt a message.

Old-time cryptography

In the olden days, Alice and Bob would have come up with some kind of system depending on a shared secret *key* with which you could encrypt and decrypt a message. Perhaps you've seen many of these techniques already!

Old-time cryptography

In the olden days, Alice and Bob would have come up with some kind of system depending on a shared secret *key* with which you could encrypt and decrypt a message. Perhaps you've seen many of these techniques already! For example, you could substitute the letters of the alphabet according to some agreed system: then the key would describe that system and would be a list of facts like $A \mapsto Q$, $B \mapsto J$, etc.

Old-time cryptography

In the olden days, Alice and Bob would have come up with some kind of system depending on a shared secret *key* with which you could encrypt and decrypt a message. Perhaps you've seen many of these techniques already! For example, you could substitute the letters of the alphabet according to some agreed system: then the key would describe that system and would be a list of facts like $A \mapsto Q$, $B \mapsto J$, etc.

The big disadvantage with that is that Alice and Bob have to exchange the key somehow at the beginning:

Old-time cryptography

In the olden days, Alice and Bob would have come up with some kind of system depending on a shared secret *key* with which you could encrypt and decrypt a message. Perhaps you've seen many of these techniques already! For example, you could substitute the letters of the alphabet according to some agreed system: then the key would describe that system and would be a list of facts like $A \mapsto Q$, $B \mapsto J$, etc.

The big disadvantage with that is that Alice and Bob have to exchange the key somehow at the beginning: if Eve can spy on that conversation, she has the key and can decrypt Alice's message just as easily as Bob can.

Old-time cryptography

In the olden days, Alice and Bob would have come up with some kind of system depending on a shared secret *key* with which you could encrypt and decrypt a message. Perhaps you've seen many of these techniques already! For example, you could substitute the letters of the alphabet according to some agreed system: then the key would describe that system and would be a list of facts like $A \mapsto Q$, $B \mapsto J$, etc.

The big disadvantage with that is that Alice and Bob have to exchange the key somehow at the beginning: if Eve can spy on that conversation, she has the key and can decrypt Alice's message just as easily as Bob can.

The problem with this old-time approach is that the same secret is used to encrypt and decrypt the message, so needs exchanging.

Public-key cryptography

Public-key cryptography

Suppose instead there was a type of encryption with a key for encryption and another key for decryption, such that, even if you know exactly how to encrypt a message, it is very hard indeed to work out how to decrypt it.

Public-key cryptography

Suppose instead there was a type of encryption with a key for encryption and another key for decryption, such that, even if you know exactly how to encrypt a message, it is very hard indeed to work out how to decrypt it.

That suggests the following plan:

Public-key cryptography

Suppose instead there was a type of encryption with a key for encryption and another key for decryption, such that, even if you know exactly how to encrypt a message, it is very hard indeed to work out how to decrypt it.

That suggests the following plan:

1. Bob comes up with a system of encrypting and decrypting of that sort.

Public-key cryptography

Suppose instead there was a type of encryption with a key for encryption and another key for decryption, such that, even if you know exactly how to encrypt a message, it is very hard indeed to work out how to decrypt it.

That suggests the following plan:

1. Bob comes up with a system of encrypting and decrypting of that sort.
2. Bob takes the key which tells you how to decrypt messages, the *private key*, and keeps it to himself, never telling anyone.

Public-key cryptography

Suppose instead there was a type of encryption with a key for encryption and another key for decryption, such that, even if you know exactly how to encrypt a message, it is very hard indeed to work out how to decrypt it.

That suggests the following plan:

1. Bob comes up with a system of encrypting and decrypting of that sort.
2. Bob takes the key which tells you how to decrypt messages, the *private key*, and keeps it to himself, never telling anyone.
3. Bob takes the key which tells you how to encrypt messages, the *public key*, and shares it with everyone who wants it, with no secrecy whatsoever. In particular, he sends Alice a postcard telling her his public key. Of course Eve finds it out quickly, but Bob doesn't care.

Public-key cryptography

Suppose instead there was a type of encryption with a key for encryption and another key for decryption, such that, even if you know exactly how to encrypt a message, it is very hard indeed to work out how to decrypt it.

That suggests the following plan:

1. Bob comes up with a system of encrypting and decrypting of that sort.
2. Bob takes the key which tells you how to decrypt messages, the *private key*, and keeps it to himself, never telling anyone.
3. Bob takes the key which tells you how to encrypt messages, the *public key*, and shares it with everyone who wants it, with no secrecy whatsoever. In particular, he sends Alice a postcard telling her his public key. Of course Eve finds it out quickly, but Bob doesn't care.
4. Alice uses Bob's public key to encrypt a message for Bob.

Public-key cryptography

Suppose instead there was a type of encryption with a key for encryption and another key for decryption, such that, even if you know exactly how to encrypt a message, it is very hard indeed to work out how to decrypt it.

That suggests the following plan:

1. Bob comes up with a system of encrypting and decrypting of that sort.
2. Bob takes the key which tells you how to decrypt messages, the *private key*, and keeps it to himself, never telling anyone.
3. Bob takes the key which tells you how to encrypt messages, the *public key*, and shares it with everyone who wants it, with no secrecy whatsoever. In particular, he sends Alice a postcard telling her his public key. Of course Eve finds it out quickly, but Bob doesn't care.
4. Alice uses Bob's public key to encrypt a message for Bob.
5. Alice sends Bob the encrypted message.

Public-key cryptography

Suppose instead there was a type of encryption with a key for encryption and another key for decryption, such that, even if you know exactly how to encrypt a message, it is very hard indeed to work out how to decrypt it.

That suggests the following plan:

1. Bob comes up with a system of encrypting and decrypting of that sort.
2. Bob takes the key which tells you how to decrypt messages, the *private key*, and keeps it to himself, never telling anyone.
3. Bob takes the key which tells you how to encrypt messages, the *public key*, and shares it with everyone who wants it, with no secrecy whatsoever. In particular, he sends Alice a postcard telling her his public key. Of course Eve finds it out quickly, but Bob doesn't care.
4. Alice uses Bob's public key to encrypt a message for Bob.
5. Alice sends Bob the encrypted message.
6. Bob uses his private key to decrypt it, and read Alice's message.

The details

The details

So the only question is, how can we come up with such a system, where being able to encrypt things doesn't help you decrypt things very much?

The details

So the only question is, how can we come up with such a system, where being able to encrypt things doesn't help you decrypt things very much?

The approach we'll describe was the first one to be thought of, in the 1970s.

The details

So the only question is, how can we come up with such a system, where being able to encrypt things doesn't help you decrypt things very much?

The approach we'll describe was the first one to be thought of, in the 1970s. It is known as *RSA* after its American inventors Rivest, Shamir and Adleman.

The details

So the only question is, how can we come up with such a system, where being able to encrypt things doesn't help you decrypt things very much?

The approach we'll describe was the first one to be thought of, in the 1970s. It is known as *RSA* after its American inventors Rivest, Shamir and Adleman. (A British mathematician, Cocks, invented it a few years earlier, but he was working in secret for the government, so this was not known for many years).

The details

So the only question is, how can we come up with such a system, where being able to encrypt things doesn't help you decrypt things very much?

The approach we'll describe was the first one to be thought of, in the 1970s. It is known as *RSA* after its American inventors Rivest, Shamir and Adleman. (A British mathematician, Cocks, invented it a few years earlier, but he was working in secret for the government, so this was not known for many years). RSA is still in very widespread use on the internet.

The details

So the only question is, how can we come up with such a system, where being able to encrypt things doesn't help you decrypt things very much?

The approach we'll describe was the first one to be thought of, in the 1970s. It is known as *RSA* after its American inventors Rivest, Shamir and Adleman. (A British mathematician, Cocks, invented it a few years earlier, but he was working in secret for the government, so this was not known for many years). RSA is still in very widespread use on the internet.

The secret of RSA is to work modulo pq, where p and q are (different) primes.

The details

So the only question is, how can we come up with such a system, where being able to encrypt things doesn't help you decrypt things very much?

The approach we'll describe was the first one to be thought of, in the 1970s. It is known as *RSA* after its American inventors Rivest, Shamir and Adleman. (A British mathematician, Cocks, invented it a few years earlier, but he was working in secret for the government, so this was not known for many years). RSA is still in very widespread use on the internet.

The secret of RSA is to work modulo pq, where p and q are (different) primes. We're going to need to do modular arithmetic mod pq, including exponentiation.

The details

So the only question is, how can we come up with such a system, where being able to encrypt things doesn't help you decrypt things very much?

The approach we'll describe was the first one to be thought of, in the 1970s. It is known as *RSA* after its American inventors Rivest, Shamir and Adleman. (A British mathematician, Cocks, invented it a few years earlier, but he was working in secret for the government, so this was not known for many years). RSA is still in very widespread use on the internet.

The secret of RSA is to work modulo pq, where p and q are (different) primes. We're going to need to do modular arithmetic mod pq, including exponentiation. So we'll need to see what Fermat-Euler says:

Fermat-Euler for pq

Fermat-Euler for *pq*

Proposition

Let p and q be different primes. Then the number $\varphi(pq)$, of integers from 1 to pq coprime to pq, is given by

$$\varphi(pq) = (p-1)(q-1).$$

Fermat-Euler for *pq*

Proposition

Let p and q be different primes. Then the number $\varphi(pq)$, of integers from 1 to pq coprime to pq, is given by

$$\varphi(pq) = (p-1)(q-1).$$

Proof.

There are pq integers a between 1 and pq. Those that are not coprime to pq are either multiples of p or of q.

Fermat-Euler for *pq*

Proposition

Let p and q be different primes. Then the number $\varphi(pq)$, of integers from 1 to pq coprime to pq, is given by

$$\varphi(pq) = (p-1)(q-1).$$

Proof.

There are pq integers a between 1 and pq. Those that are not coprime to pq are either multiples of p or of q.
Of these, q of them are multiples of p (namely $p, 2p, \ldots, pq$).

Fermat-Euler for *pq*

Proposition

Let p and q be different primes. Then the number $\varphi(pq)$, of integers from 1 to pq coprime to pq, is given by

$$\varphi(pq) = (p-1)(q-1).$$

Proof.

There are pq integers a between 1 and pq. Those that are not coprime to pq are either multiples of p or of q.
Of these, q of them are multiples of p (namely $p, 2p, \ldots, pq$).
Also, p of them are multiples of q (namely $q, 2q, \ldots, pq$).

Fermat-Euler for *pq*

Proposition

Let p and q be different primes. Then the number $\varphi(pq)$, of integers from 1 to pq coprime to pq, is given by

$$\varphi(pq) = (p-1)(q-1).$$

Proof.

There are pq integers a between 1 and pq. Those that are not coprime to pq are either multiples of p or of q.
Of these, q of them are multiples of p (namely $p, 2p, \ldots, pq$).
Also, p of them are multiples of q (namely $q, 2q, \ldots, pq$).
Lastly, one of them (namely pq) is a multiple of p and of q.

Fermat-Euler for *pq*

Proposition

Let p and q be different primes. Then the number $\varphi(pq)$, of integers from 1 to pq coprime to pq, is given by

$$\varphi(pq) = (p-1)(q-1).$$

Proof.

There are pq integers a between 1 and pq. Those that are not coprime to pq are either multiples of p or of q.
Of these, q of them are multiples of p (namely $p, 2p, \ldots, pq$).
Also, p of them are multiples of q (namely $q, 2q, \ldots, pq$).
Lastly, one of them (namely pq) is a multiple of p and of q.
Hence $q + p - 1$ are not coprime to pq,

Fermat-Euler for *pq*

Proposition

Let p and q be different primes. Then the number $\varphi(pq)$, of integers from 1 *to pq coprime to pq, is given by*

$$\varphi(pq) = (p-1)(q-1).$$

Proof.

There are pq integers a between 1 and pq. Those that are not coprime to pq are either multiples of p or of q.
Of these, q of them are multiples of p (namely $p, 2p, \ldots, pq$).
Also, p of them are multiples of q (namely $q, 2q, \ldots, pq$).
Lastly, one of them (namely pq) is a multiple of p and of q.
Hence $q + p - 1$ are not coprime to pq, and so

$$\varphi(pq) = pq - q - p + 1 = (p-1)(q-1).$$

□

Now we know $\varphi(pq) = (p-1)(q-1)$

Now we know $\varphi(pq) = (p-1)(q-1)$

Remark

As a result of that, we know (from the Fermat-Euler Theorem that, for all a coprime to pq, we have

$$a^{(p-1)(q-1)} \equiv 1 \pmod{pq},$$

Now we know $\varphi(pq) = (p-1)(q-1)$

Remark

As a result of that, we know (from the Fermat-Euler Theorem that, for all a coprime to pq, we have

$$a^{(p-1)(q-1)} \equiv 1 \pmod{pq},$$

and indeed

$$a^{k(p-1)(q-1)} \equiv 1 \pmod{pq}.$$

for all k.

Back to Alice and Bob

Back to Alice and Bob

So, Bob chooses two fairly large primes p and q, and keeps them secret.

Back to Alice and Bob

So, Bob chooses two fairly large primes p and q, and keeps them secret. He also chooses a number e which is coprime to $(p-1)(q-1)$.

Back to Alice and Bob

So, Bob chooses two fairly large primes p and q, and keeps them secret. He also chooses a number e which is coprime to $(p-1)(q-1)$.
He also calculates the inverse d to e, modulo $(p-1)(q-1)$, by using Euclid's algorithm.

Back to Alice and Bob

So, Bob chooses two fairly large primes p and q, and keeps them secret. He also chooses a number e which is coprime to $(p-1)(q-1)$.

He also calculates the inverse d to e, modulo $(p-1)(q-1)$, by using Euclid's algorithm.

His public key consists of pq and e, so he sends that to Alice (and Eve);

Back to Alice and Bob

So, Bob chooses two fairly large primes p and q, and keeps them secret. He also chooses a number e which is coprime to $(p-1)(q-1)$.
He also calculates the inverse d to e, modulo $(p-1)(q-1)$, by using Euclid's algorithm.
His public key consists of pq and e, so he sends that to Alice (and Eve); his private key consists of pq and d.

Back to Alice and Bob

So, Bob chooses two fairly large primes p and q, and keeps them secret. He also chooses a number e which is coprime to $(p-1)(q-1)$.
He also calculates the inverse d to e, modulo $(p-1)(q-1)$, by using Euclid's algorithm.
His public key consists of pq and e, so he sends that to Alice (and Eve); his private key consists of pq and d. He shreds any evidence of what p and q are.

Back to Alice and Bob

So, Bob chooses two fairly large primes p and q, and keeps them secret. He also chooses a number e which is coprime to $(p-1)(q-1)$.

He also calculates the inverse d to e, modulo $(p-1)(q-1)$, by using Euclid's algorithm.

His public key consists of pq and e, so he sends that to Alice (and Eve); his private key consists of pq and d. He shreds any evidence of what p and q are.

Alice represents her message as a number m between 1 and pq.

Back to Alice and Bob

So, Bob chooses two fairly large primes p and q, and keeps them secret. He also chooses a number e which is coprime to $(p-1)(q-1)$.

He also calculates the inverse d to e, modulo $(p-1)(q-1)$, by using Euclid's algorithm.

His public key consists of pq and e, so he sends that to Alice (and Eve); his private key consists of pq and d. He shreds any evidence of what p and q are.

Alice represents her message as a number m between 1 and pq. It is overwhelmingly likely that her choice will be coprime to pq.

Back to Alice and Bob

So, Bob chooses two fairly large primes p and q, and keeps them secret. He also chooses a number e which is coprime to $(p-1)(q-1)$.

He also calculates the inverse d to e, modulo $(p-1)(q-1)$, by using Euclid's algorithm.

His public key consists of pq and e, so he sends that to Alice (and Eve); his private key consists of pq and d. He shreds any evidence of what p and q are.

Alice represents her message as a number m between 1 and pq. It is overwhelmingly likely that her choice will be coprime to pq. She calculates

$$m^e \pmod{pq}$$

and sends it on to Bob.

Alice and Bob continued

Alice and Bob continued

Bob receives this number m^e from Alice, and raises it to the power d modulo pq.

Alice and Bob continued

Bob receives this number m^e from Alice, and raises it to the power d modulo pq. He thus obtains something congruent to

$$(m^e)^d = m^{de}.$$

Alice and Bob continued

Bob receives this number m^e from Alice, and raises it to the power d modulo pq. He thus obtains something congruent to

$$(m^e)^d = m^{de}.$$

Now, because $de \equiv 1 \pmod{\varphi(pq)}$, we have $de = 1 + k\varphi(pq)$ for some k.

Alice and Bob continued

Bob receives this number m^e from Alice, and raises it to the power d modulo pq. He thus obtains something congruent to

$$(m^e)^d = m^{de}.$$

Now, because $de \equiv 1 \pmod{\varphi(pq)}$, we have $de = 1 + k\varphi(pq)$ for some k. As a result,

$$(m^e)^d$$

Alice and Bob continued

Bob receives this number m^e from Alice, and raises it to the power d modulo pq. He thus obtains something congruent to

$$(m^e)^d = m^{de}.$$

Now, because $de \equiv 1 \pmod{\varphi(pq)}$, we have $de = 1 + k\varphi(pq)$ for some k. As a result,

$$(m^e)^d = m^{de}$$

Alice and Bob continued

Bob receives this number m^e from Alice, and raises it to the power d modulo pq. He thus obtains something congruent to

$$(m^e)^d = m^{de}.$$

Now, because $de \equiv 1 \pmod{\varphi(pq)}$, we have $de = 1 + k\varphi(pq)$ for some k. As a result,

$$(m^e)^d = m^{de} = m^{1+k\varphi(pq)}$$

Alice and Bob continued

Bob receives this number m^e from Alice, and raises it to the power d modulo pq. He thus obtains something congruent to

$$(m^e)^d = m^{de}.$$

Now, because $de \equiv 1 \pmod{\varphi(pq)}$, we have $de = 1 + k\varphi(pq)$ for some k. As a result,

$$(m^e)^d = m^{de} = m^{1+k\varphi(pq)} = m(m^{\varphi(pq)})^k$$

Alice and Bob continued

Bob receives this number m^e from Alice, and raises it to the power d modulo pq. He thus obtains something congruent to

$$(m^e)^d = m^{de}.$$

Now, because $de \equiv 1 \pmod{\varphi(pq)}$, we have $de = 1 + k\varphi(pq)$ for some k. As a result,

$$(m^e)^d = m^{de} = m^{1+k\varphi(pq)} = m(m^{\varphi(pq)})^k \equiv m1^k$$

Alice and Bob continued

Bob receives this number m^e from Alice, and raises it to the power d modulo pq. He thus obtains something congruent to

$$(m^e)^d = m^{de}.$$

Now, because $de \equiv 1 \pmod{\varphi(pq)}$, we have $de = 1 + k\varphi(pq)$ for some k. As a result,

$$(m^e)^d = m^{de} = m^{1+k\varphi(pq)} = m(m^{\varphi(pq)})^k \equiv m1^k \equiv m \pmod{pq}.$$

Alice and Bob continued

Bob receives this number m^e from Alice, and raises it to the power d modulo pq. He thus obtains something congruent to

$$(m^e)^d = m^{de}.$$

Now, because $de \equiv 1 \pmod{\varphi(pq)}$, we have $de = 1 + k\varphi(pq)$ for some k. As a result,

$$(m^e)^d = m^{de} = m^{1+k\varphi(pq)} = m(m^{\varphi(pq)})^k \equiv m1^k \equiv m \pmod{pq}.$$

Hence, using his private key, Bob can recover what m was from being told m^e.

Security

Security

The idea is that it should be very hard for anyone else to work out d from pq and e;

Security

The idea is that it should be very hard for anyone else to work out d from pq and e; we did this using Euclid's algorithm, but we needed to know more than just pq: we needed to know $(p-1)(q-1)$.

Security

The idea is that it should be very hard for anyone else to work out d from pq and e; we did this using Euclid's algorithm, but we needed to know more than just pq: we needed to know $(p-1)(q-1)$. So the security of this approach depends (among other things) on it being difficult to factorise the number pq:

Security

The idea is that it should be very hard for anyone else to work out d from pq and e; we did this using Euclid's algorithm, but we needed to know more than just pq: we needed to know $(p-1)(q-1)$. So the security of this approach depends (among other things) on it being difficult to factorise the number pq: if factorising large numbers were easy, we could get p and q for ourselves from Bob's public key.

Security

The idea is that it should be very hard for anyone else to work out d from pq and e; we did this using Euclid's algorithm, but we needed to know more than just pq: we needed to know $(p-1)(q-1)$. So the security of this approach depends (among other things) on it being difficult to factorise the number pq: if factorising large numbers were easy, we could get p and q for ourselves from Bob's public key. Currently, we know of no way to do this fast enough: we know how to generate primes that are hundreds of digits long, but not to factorise a product of two of them.