Cryptography: Public Keys and Euclidean Algorithm

Alex Corner

Sheffield Hallam University

The Story So Far

- We've looked at the encryption standards DES and AES. DES is now insecure, but AES is very much in use and believed to be uncrackable for the time being.

The Story So Far

- We've looked at the encryption standards DES and AES. DES is now insecure, but AES is very much in use and believed to be uncrackable for the time being.
- Even with the most secure cryptographic algorithms, we still have a problem that we haven't dealt with yet:

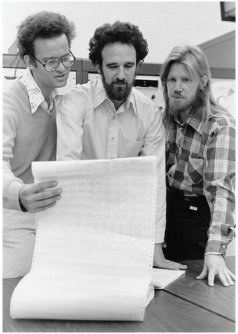
The Story So Far

- We've looked at the encryption standards DES and AES. DES is now insecure, but AES is very much in use and believed to be uncrackable for the time being.
- Even with the most secure cryptographic algorithms, we still have a problem that we haven't dealt with yet: how do we securely transmit keys in order to use these algorithms?

The Story So Far

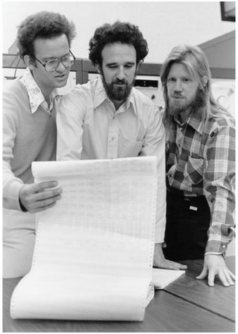
- We've looked at the encryption standards DES and AES. DES is now insecure, but AES is very much in use and believed to be uncrackable for the time being.
- Even with the most secure cryptographic algorithms, we still have a problem that we haven't dealt with yet: how do we securely transmit keys in order to use these algorithms?
- This is where **public key cryptography** comes into play.

Diffie–Hellman Key Exchange



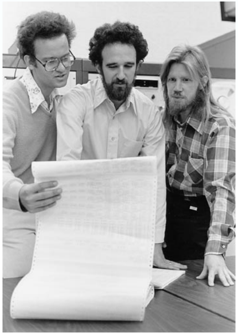
- Alice and Bob want to securely communicate using AES-128, but are having trouble with securely communicating a shared private key.

Diffie–Hellman Key Exchange



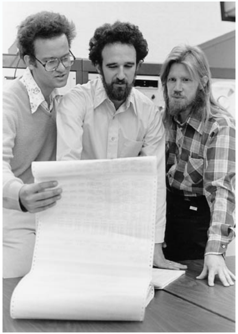
- Alice and Bob want to securely communicate using AES-128, but are having trouble with securely communicating a shared private key.
- Ralph Merkle has the solution: public key exchange.

Diffie–Hellman Key Exchange



- Alice and Bob want to securely communicate using AES-128, but are having trouble with securely communicating a shared private key.
- Ralph Merkle has the solution: public key exchange.
- The idea was communicated by Whitfield Diffie and Martin Hellman in 1976.

Diffie–Hellman–Merkle Key Exchange



- Alice and Bob want to securely communicate using AES-128, but are having trouble with securely communicating a shared private key.
- Ralph Merkle has the solution: public key exchange.
- The idea was communicated by Whitfield Diffie and Martin Hellman in 1976.

Diffie–Hellman–Merkle Key Exchange

- In 1997 the British Government declassified documents pertaining to public key exchange protocols.

Diffie–Hellman–Merkle Key Exchange

- In 1997 the British Government declassified documents pertaining to public key exchange protocols.
- James Ellis had similar thoughts about key exchange and these were further developed by Clifford Cocks and Malcolm Williamson (pictured) while they worked at GCHQ.

Diffie–Hellman–Merkle Key Exchange

- In 1997 the British Government declassified documents pertaining to public key exchange protocols.
- James Ellis had similar thoughts about key exchange and these were further developed by Clifford Cocks and Malcolm Williamson (pictured) while they worked at GCHQ.
- This type of story is common in the history of cryptography.

Diffie–Hellman–Merkle–Ellis–Cocks–Williamson Key Exchange

- In 1997 the British Government declassified documents pertaining to public key exchange protocols.
- James Ellis had similar thoughts about key exchange and these were further developed by Clifford Cocks and Malcolm Williamson (pictured) while they worked at GCHQ.
- This type of story is common in the history of cryptography.

Diffie–Hellman Key Exchange: Implementation

- Alice and Bob want to securely communicate using AES-128, but are having trouble with securely communicating a shared private key.

Diffie–Hellman Key Exchange: Implementation

- Alice and Bob want to securely communicate using AES-128, but are having trouble with securely communicating a shared private key.
- Diffie and Hellman articulated a version of Merkle's idea as follows:

Diffie–Hellman Key Exchange: Implementation

- Alice and Bob want to securely communicate using AES-128, but are having trouble with securely communicating a shared private key.
- Diffie and Hellman articulated a version of Merkle's idea as follows:
  - Alice and Bob agree on a prime number p and a generator α.
  - Their calculations are performed modulo p.
  - Alice secretly chooses a number x and Bob secretly chooses a number y.

Diffie–Hellman Key Exchange: Implementation

- Alice and Bob want to securely communicate using AES-128, but are having trouble with securely communicating a shared private key.
- Diffie and Hellman articulated a version of Merkle's idea as follows:
  - Alice and Bob agree on a prime number p and a generator α.
  - Their calculations are performed modulo p.
  - Alice secretly chooses a number x and Bob secretly chooses a number y.
  - Alice calculates $\alpha^x \mod p$ and Bob calculates $\alpha^y \mod p$: these values are then exchanged.

Diffie–Hellman Key Exchange: Implementation

- Alice and Bob want to securely communicate using AES-128, but are having trouble with securely communicating a shared private key.
- Diffie and Hellman articulated a version of Merkle's idea as follows:
  - Alice and Bob agree on a prime number p and a generator α.
  - Their calculations are performed modulo p.
  - Alice secretly chooses a number x and Bob secretly chooses a number y.
  - Alice calculates $\alpha^x \mod p$ and Bob calculates $\alpha^y \mod p$: these values are then exchanged.
  - Alice then finds $(\alpha^y)^x \mod p$ and Bob finds $(\alpha^x)^y \mod p$: this means that Alice and Bob now both have a number $\alpha^{xy} \mod p$ without ever sharing their secret values x and y.

Diffie–Hellman Key Exchange: Example

- Suppose that Alice and Bob agree to use the prime $p = 7$ and the generator $\alpha = 3$.

Diffie–Hellman Key Exchange: Example

- Suppose that Alice and Bob agree to use the prime $p = 7$ and the generator $\alpha = 3$.
- Now suppose Alice chooses $x = 2$ and Bob chooses $y = 5$.

Diffie–Hellman Key Exchange: Example

- Suppose that Alice and Bob agree to use the prime $p = 7$ and the generator $\alpha = 3$.
- Now suppose Alice chooses $x = 2$ and Bob chooses $y = 5$.
- Alice calculates $\alpha^x = 3^2 = 9 = 2 \bmod 7$ and Bob calculates $\alpha^y = 3^5 = 243 = 5 \bmod 7$.

Diffie–Hellman Key Exchange: Example

- Suppose that Alice and Bob agree to use the prime $p = 7$ and the generator $\alpha = 3$.
- Now suppose Alice chooses $x = 2$ and Bob chooses $y = 5$.
- Alice calculates $\alpha^x = 3^2 = 9 = 2 \bmod 7$ and Bob calculates $\alpha^y = 3^5 = 243 = 5 \bmod 7$.
- Alice and Bob exchange these values.

Diffie–Hellman Key Exchange: Example

- Suppose that Alice and Bob agree to use the prime $p = 7$ and the generator $\alpha = 3$.
- Now suppose Alice chooses $x = 2$ and Bob chooses $y = 5$.
- Alice calculates $\alpha^x = 3^2 = 9 = 2 \bmod 7$ and Bob calculates $\alpha^y = 3^5 = 243 = 5 \bmod 7$.
- Alice and Bob exchange these values.
- Alice finds
$$(\alpha^y)^x = 5^2 = 25 = 4 \bmod 7.$$

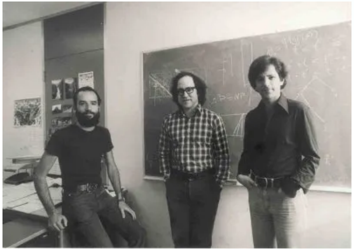
Diffie–Hellman Key Exchange: Example

- Suppose that Alice and Bob agree to use the prime $p = 7$ and the generator $\alpha = 3$.
- Now suppose Alice chooses $x = 2$ and Bob chooses $y = 5$.
- Alice calculates $\alpha^x = 3^2 = 9 = 2 \bmod 7$ and Bob calculates $\alpha^y = 3^5 = 243 = 5 \bmod 7$.
- Alice and Bob exchange these values.
- Alice finds
$$(\alpha^y)^x = 5^2 = 25 = 4 \bmod 7.$$
- Bob finds
$$(\alpha^x)^y = 2^5 = 32 = 4 \bmod 7.$$

Diffie–Hellman Key Exchange: Example

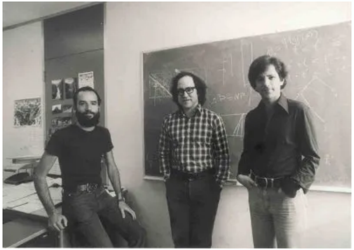
- Suppose that Alice and Bob agree to use the prime $p = 7$ and the generator $\alpha = 3$.
- Now suppose Alice chooses $x = 2$ and Bob chooses $y = 5$.
- Alice calculates $\alpha^x = 3^2 = 9 = 2 \bmod 7$ and Bob calculates $\alpha^y = 3^5 = 243 = 5 \bmod 7$.
- Alice and Bob exchange these values.
- Alice finds
$$(\alpha^y)^x = 5^2 = 25 = 4 \bmod 7.$$
- Bob finds
$$(\alpha^x)^y = 2^5 = 32 = 4 \bmod 7.$$
- Alice and Bob now share a secret value and can use this to generate their private key.

RSA: Development



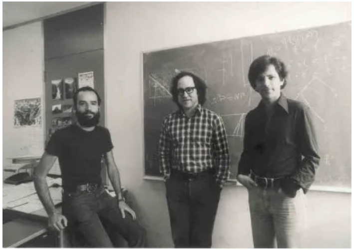
- Key exchange was a great innovation, but another idea about public key cryptography was quick to follow.

RSA: Development



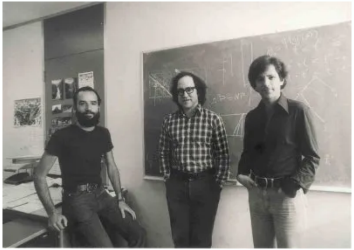
- Key exchange was a great innovation, but another idea about public key cryptography was quick to follow.
- What if we let everybody know the encryption key, but kept another key secret which was only used for decryption?

RSA: Development



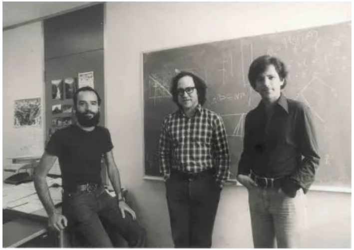
- Key exchange was a great innovation, but another idea about public key cryptography was quick to follow.
- What if we let everybody know the encryption key, but kept another key secret which was only used for decryption?
- Various cryptographers had this idea and knew it would rely on finding a mathematical function which was **one-way**: many inputs could produce the same output.

RSA: Development



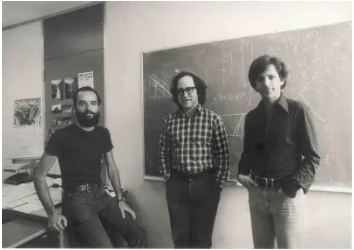
- RSA was first publicly described in 1977: United States Patent US4405829A. The authors were Ron Rivest, Adi Shamir, and Len Adleman, working at MIT at the time.

RSA: Development



- RSA was first publicly described in 1977: United States Patent US4405829A. The authors were Ron Rivest, Adi Shamir, and Len Adleman, working at MIT at the time.
- (An equivalent method was developed by Clifford Cocks in 1973, again at GCHQ, but this wasn't declassified until 1997.)

RSA: Development



- RSA was first publicly described in 1977: United States Patent US4405829A. The authors were Ron Rivest, Adi Shamir, and Len Adleman, working at MIT at the time.
- (An equivalent method was developed by Clifford Cocks in 1973, again at GCHQ, but this wasn't declassified until 1997.)
- The scheme depends on large prime numbers, modular arithmetic, and the inherent difficulty in factorising large products.

Some Number Theory

- Let p be a **prime number**: a number which is divisible only by itself and by 1.

Some Number Theory

- Let p be a **prime number**: a number which is divisible only by itself and by 1.
- We do not include the number 1 in this definition.

Some Number Theory

- Let p be a **prime number**: a number which is divisible only by itself and by 1.
- We do not include the number 1 in this definition.
- The first few primes are 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, ...

Some Number Theory

- Let p be a **prime number**: a number which is divisible only by itself and by 1.
- We do not include the number 1 in this definition.
- The first few primes are 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, ...
- An integer a is **coprime** to p if the **highest common factor** of a and p is 1: $\mathrm{hcf}(a, p) = 1$.

Some Number Theory

- Let p be a **prime number**: a number which is divisible only by itself and by 1.
- We do not include the number 1 in this definition.
- The first few primes are $2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, \ldots$.
- An integer a is **coprime** to p if the **highest common factor** of a and p is 1: $\mathrm{hcf}(a, p) = 1$. Also known as the **greatest common divisor**.

Some Number Theory

- Let p be a **prime number**: a number which is divisible only by itself and by 1.
- We do not include the number 1 in this definition.
- The first few primes are 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, ...
- An integer a is **coprime** to p if the **highest common factor** of a and p is 1: $\mathrm{hcf}(a, p) = 1$. Also known as the **greatest common divisor**.
- E.g., $\mathrm{hcf}(105, 195) = 15$ since $105 = 3 \times 5 \times 7$ and $195 = 3 \times 5 \times 13$.

Some Number Theory

- Let p be a **prime number**: a number which is divisible only by itself and by 1.
- We do not include the number 1 in this definition.
- The first few primes are 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, ...
- An integer a is **coprime** to p if the **highest common factor** of a and p is 1: $\text{hcf}(a, p) = 1$. Also known as the **greatest common divisor**.
- E.g., $\text{hcf}(105, 195) = 15$ since $105 = 3 \times 5 \times 7$ and $195 = 3 \times 5 \times 13$.
- Let n be a non-negative integer. The number of integers less than n which are coprime to n is written as $\varphi(n)$. The function φ is known as **Euler's totient function**.

Some Number Theory

- Let p be a **prime number**: a number which is divisible only by itself and by 1.
- We do not include the number 1 in this definition.
- The first few primes are 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, ...
- An integer a is **coprime** to p if the **highest common factor** of a and p is 1: $\text{hcf}(a, p) = 1$. Also known as the **greatest common divisor**.
- E.g., $\text{hcf}(105, 195) = 15$ since $105 = 3 \times 5 \times 7$ and $195 = 3 \times 5 \times 13$.
- Let n be a non-negative integer. The number of integers less than n which are coprime to n is written as $\varphi(n)$. The function φ is known as **Euler's totient function**.
- E.g., $\varphi(6) = 2$ since only 1 and 5 are less than 6 and coprime to 6. Also $\varphi(10) = 4$: 1, 3, 7, and 9 are coprime to 10.

Some Number Theory

- Let p be a **prime number**: a number which is divisible only by itself and by 1.
- We do not include the number 1 in this definition.
- The first few primes are 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, ...
- An integer a is **coprime** to p if the **highest common factor** of a and p is 1: $\text{hcf}(a, p) = 1$. Also known as the **greatest common divisor**.
- E.g., $\text{hcf}(105, 195) = 15$ since $105 = 3 \times 5 \times 7$ and $195 = 3 \times 5 \times 13$.
- Let n be a non-negative integer. The number of integers less than n which are coprime to n is written as $\varphi(n)$. The function φ is known as **Euler's totient function**.
- E.g., $\varphi(6) = 2$ since only 1 and 5 are less than 6 and coprime to 6. Also $\varphi(10) = 4$: 1, 3, 7, and 9 are coprime to 10.
- This is easy to compute for prime numbers: $\varphi(p) = p - 1$. And for products of primes: $\varphi(pq) = (p - 1)(q - 1)$.

RSA: Implementation

- We'll look at how RSA actually works next week!

Euclidean Algorithm

How could we easily check that the highest common factor of $e = 5$ and $a = 72$ is actually equal to 1?

Euclidean Algorithm

How could we easily check that the highest common factor of $e = 5$ and $a = 72$ is actually equal to 1?

$$
\begin{aligned}
72 &= 14 \times 5 + 2 \\
5 &= 2 \times 2 + 1 \\
2 &= 2 \times 1 + 0
\end{aligned}
$$

Euclidean Algorithm

How could we easily check that the highest common factor of $e = 5$ and $a = 72$ is actually equal to 1?

$$
\begin{aligned}
72 &= 14 \times 5 + 2 \\
5 &= 2 \times 2 + 1 \\
2 &= 2 \times 1 + 0
\end{aligned}
$$

The least number before 0 is the highest common factor. Here this is 1, so e and a are coprime.

Euclidean Algorithm: Example

Here's another example of the Euclidean Algorithm to show that $\mathrm{hcf}(72, 26) = 2$.

$$\begin{aligned} 72 &= 2 \times 26 + 20 \\ 26 &= 1 \times 20 + 6 \\ 20 &= 3 \times 6 + 2 \\ 6 &= 3 \times 2 + 0 \end{aligned}$$

Diophantine Equations

Often in mathematical applications we want to find solutions to equations such as

$$72a + 5b = c.$$

When we implement RSA we will need to do something similar, except we will need each of a, b, and c to be integers. These types of equations where we are interested in integer solutions are called **Diophantine equations**.

Extended Euclidean Algorithm

Extended Euclidean Algorithm

$$
\begin{array}{cc c cc}
(1,0) & 72 & & 5 & (0,1) \\
 & & \times 14 & & \\
(0,14) & 70 & & 4 & (2,-28) \\ \cline{4-5}
 & & \times 2 & & \\ \cline{1-2}
(1,-14) & 2 & & 1 & (-2,29) \\
 & & \times 2 & & \\
(-4,58) & 2 & & & \\ \cline{1-2}
(5,-72) & 0 & & &
\end{array}
$$

Extended Euclidean Algorithm

$$
\begin{array}{ccccc}
(1,0) & 72 & & 5 & (0,1) \\
 & & \times 14 & & \\
(0,14) & 70 & & 4 & (2,-28) \\
\cline{4-5}
 & & \times 2 & & \\
\cline{1-2}
(1,-14) & 2 & & 1 & (-2,29) \\
 & & \times 2 & & \\
(-4,58) & 2 & & & \\
\cline{1-2}
(5,-72) & 0 & & &
\end{array}
$$

At each stage, the pair of coordinates (a, b) corresponding to a number c in the table tells us that $72a + 5b = c$. E.g., the pair $(2, -28)$ corresponds to the number 4 in the table, so we know that $72 \times 2 + 5 \times (-28) = 4$.

Extended Euclidean Algorithm

$$
\begin{array}{cccccc}
(1,0) & 72 & & 5 & (0,1) \\
 & & \times 14 & & \\
(0,14) & 70 & & 4 & (2,-28) \\
\cline{4-5}
 & & \times 2 & & \\
\hline
(1,-14) & 2 & & 1 & (-2,29) \\
 & & \times 2 & & \\
(-4,58) & 2 & & & \\
\hline
(5,-72) & 0 & & &
\end{array}
$$

At each stage, the pair of coordinates (a, b) corresponding to a number c in the table tells us that $72a + 5b = c$. E.g., the pair $(2, -28)$ corresponds to the number 4 in the table, so we know that $72 \times 2 + 5 \times (-28) = 4$. Similarly, we know that $72 \times (-2) + 5 \times 29 = 1$.

Extended Euclidean Algorithm

$$
\begin{array}{ccccc}
(1,0) & 72 & & 5 & (0,1) \\
 & & \times 14 & & \\
(0,14) & 70 & & 4 & (2,-28) \\
\cline{4-5}
 & & \times 2 & & \\
\cline{1-2}
(1,-14) & 2 & & 1 & (-2,29) \\
 & & \times 2 & & \\
(-4,58) & 2 & & & \\
\cline{1-2}
(5,-72) & 0 & & &
\end{array}
$$

At each stage, the pair of coordinates (a, b) corresponding to a number c in the table tells us that $72a + 5b = c$. E.g., the pair $(2, -28)$ corresponds to the number 4 in the table, so we know that $72 \times 2 + 5 \times (-28) = 4$. Similarly, we know that $72 \times (-2) + 5 \times 29 = 1$.
There is only ever a solution to $xa + yb = c$ if c is a multiple of $\text{hcf}(x, y)$.

Extended Euclidean Algorithm: Example

Are there any solutions to the equation $180x + 7y = 1$?

Extended Euclidean Algorithm: Example

Are there any solutions to the equation $180x + 7y = 1$?

$$
\begin{array}{cc c cc}
(1,0) & 180 & & 7 & (0,1) \\
 & & \times 25 & & \\
(0,25) & 175 & & 5 & (1,-25) \\
\cline{4-5}
 & & \times 1 & & \\
\hline
(1,-25) & 5 & & 2 & (-1,26) \\
 & & \times 2 & & \\
(-2,52) & 4 & & & \\
\hline
(3,-77) & 1 & & &
\end{array}
$$

Extended Euclidean Algorithm: Example

Are there any solutions to the equation $180x + 7y = 1$?

$$
\begin{array}{ccccc}
(1,0) & 180 & & 7 & (0,1) \\
 & & \times 25 & & \\
(0,25) & 175 & & 5 & (1,-25) \\
\cline{4-5}
 & & \times 1 & & \\
\cline{1-2}
(1,-25) & 5 & & 2 & (-1,26) \\
 & & \times 2 & & \\
(-2,52) & 4 & & & \\
\cline{1-2}
(3,-77) & 1 & & &
\end{array}
$$

Yes!

Extended Euclidean Algorithm: Example

Are there any solutions to the equation $180x + 7y = 1$?

$$
\begin{array}{ccccc}
(1,0) & 180 & & 7 & (0,1) \\
 & & \times 25 & & \\
(0,25) & 175 & & 5 & (1,-25) \\
\cline{4-5}
 & & \times 1 & & \\
\hline
(1,-25) & 5 & & 2 & (-1,26) \\
 & & \times 2 & & \\
(-2,52) & 4 & & & \\
\cline{1-2}
(3,-77) & 1 & & &
\end{array}
$$

Yes! Take $x = 3$ and $y = -77$. Then

$$180 \times 3 + 7 \times (-77) = 1.$$

Extended Euclidean Algorithm: Examples

Let's do some more examples!

Extended Euclidean Algorithm: Examples

A board game company is designing a game. Each copy of the game includes 1204 tokens. There are two types of box which are used to store the tokens, the type A boxes can hold 42 tokens and the type B boxes can hold 119 tokens.

1. The company wants to order boxes for each game in order to store the tokens. That is, they want to order a number of type A boxes and a number of type B boxes which will exactly hold the 1204 tokens of the game. Formulate this problem in the form $sa + tb = n$, where $a \in \mathbb{Z}$ is the number of type A boxes, $b \in \mathbb{Z}$ is the number of type B boxes, n is the total number of tokens in the board game, and $s, t \in \mathbb{Z}$.

Extended Euclidean Algorithm: Examples

A board game company is designing a game. Each copy of the game includes 1204 tokens. There are two types of box which are used to store the tokens, the type A boxes can hold 42 tokens and the type B boxes can hold 119 tokens.

1. The company wants to order boxes for each game in order to store the tokens. That is, they want to order a number of type A boxes and a number of type B boxes which will exactly hold the 1204 tokens of the game. Formulate this problem in the form $sa + tb = n$, where $a \in \mathbb{Z}$ is the number of type A boxes, $b \in \mathbb{Z}$ is the number of type B boxes, n is the total number of tokens in the board game, and $s, t \in \mathbb{Z}$.
2. What further restriction should we put on a and b in order for this problem to make sense?

Extended Euclidean Algorithm: Examples

A board game company is designing a game. Each copy of the game includes 1204 tokens. There are two types of box which are used to store the tokens, the type A boxes can hold 42 tokens and the type B boxes can hold 119 tokens.

1. The company wants to order boxes for each game in order to store the tokens. That is, they want to order a number of type A boxes and a number of type B boxes which will exactly hold the 1204 tokens of the game. Formulate this problem in the form $sa + tb = n$, where $a \in \mathbb{Z}$ is the number of type A boxes, $b \in \mathbb{Z}$ is the number of type B boxes, n is the total number of tokens in the board game, and $s, t \in \mathbb{Z}$.
2. What further restriction should we put on a and b in order for this problem to make sense?
3. Use the Extended Euclidean Algorithm in order to find the highest common factor of 42 and 119.

Extended Euclidean Algorithm: Examples

A board game company is designing a game. Each copy of the game includes 1204 tokens. There are two types of box which are used to store the tokens, the type A boxes can hold 42 tokens and the type B boxes can hold 119 tokens.

1. The company wants to order boxes for each game in order to store the tokens. That is, they want to order a number of type A boxes and a number of type B boxes which will exactly hold the 1204 tokens of the game. Formulate this problem in the form $sa + tb = n$, where $a \in \mathbb{Z}$ is the number of type A boxes, $b \in \mathbb{Z}$ is the number of type B boxes, n is the total number of tokens in the board game, and s, $t \in \mathbb{Z}$.
2. What further restriction should we put on a and b in order for this problem to make sense?
3. Use the Extended Euclidean Algorithm in order to find the highest common factor of 42 and 119.
4. Are there any solutions to the problem in part (a)?

Tutorials

In the tutorial this week we will:

- Create a spreadsheet to handle Extended Euclidean Algorithm calculations.
- Practice performing the Euclidean Algorithm by hand and checking it with the spreadsheet.