Discrete Mathematics & Mathematical Reasoning
Greatest Common Divisors

Colin Stirling

Informatics
## Greatest common divisor

**Definition**

Let \( a, b \in \mathbb{Z}^+ \). The largest integer \( d \) such that \( d \mid a \) and \( d \mid b \) is called the greatest common divisor of \( a \) and \( b \), written \( \gcd(a, b) \).
Greatest common divisor

**Definition**

Let \( a, b \in \mathbb{Z}^+ \). The largest integer \( d \) such that \( d \mid a \) and \( d \mid b \) is called the greatest common divisor of \( a \) and \( b \), written \( \gcd(a, b) \)

\[ \gcd(36, 24) = 12 \]

Although 9 and 22 are coprime they are both composite.

Colin Stirling (Informatics)  
Discrete Mathematics (Chap 4)
# Greatest common divisor

## Definition

Let $a, b \in \mathbb{Z}^+$. The largest integer $d$ such that $d|a$ and $d|b$ is called the greatest common divisor of $a$ and $b$, written $\gcd(a, b)$.

<table>
<thead>
<tr>
<th>$\gcd(36, 24)$</th>
<th>$12$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\gcd(22, 9)$</td>
<td>$1$</td>
</tr>
</tbody>
</table>
Greatest common divisor

Definition
Let $a, b \in \mathbb{Z}^+$. The largest integer $d$ such that $d \mid a$ and $d \mid b$ is called the greatest common divisor of $a$ and $b$, written $\text{gcd}(a, b)$.

\[
\text{gcd}(36, 24) = 12
\]

\[
\text{gcd}(22, 9) = 1
\]

Definition
The integers $a$ and $b$ are relatively prime (coprime) iff $\text{gcd}(a, b) = 1$. 
Greatest common divisor

**Definition**

Let $a, b \in \mathbb{Z}^+$. The largest integer $d$ such that $d|a$ and $d|b$ is called the greatest common divisor of $a$ and $b$, written $\gcd(a, b)$

$\gcd(36, 24) = 12$

$\gcd(22, 9) = 1$

**Definition**

The integers $a$ and $b$ are relatively prime (coprime) iff $\gcd(a, b) = 1$

Although 9 and 22 are coprime they are both composite
Gcd by prime factorisations

Suppose that the prime factorisations of $a$ and $b$ are

$$a = p_1^{a_1} p_2^{a_2} \cdots p_n^{a_n} \quad b = p_1^{b_1} p_2^{b_2} \cdots p_n^{b_n}$$

where each exponent is a nonnegative integer (possibly zero).
Gcd by prime factorisations

Suppose that the prime factorisations of \(a\) and \(b\) are

\[
a = p_1^{a_1} p_2^{a_2} \cdots p_n^{a_n} \quad b = p_1^{b_1} p_2^{b_2} \cdots p_n^{b_n}
\]

where each exponent is a nonnegative integer (possibly zero)

\[
\text{gcd}(a, b) = p_1^{\min(a_1, b_1)} p_2^{\min(a_2, b_2)} \cdots p_n^{\min(a_n, b_n)}
\]

This number clearly divides \(a\) and \(b\). No larger number can divide both \(a\) and \(b\). Proof by contradiction and the prime factorisation of a postulated larger divisor.

Factorisation is a very inefficient method to compute gcd
Gcd by prime factorisations

Suppose that the prime factorisations of $a$ and $b$ are

$$a = p_1^{a_1} p_2^{a_2} \cdots p_n^{a_n} \quad b = p_1^{b_1} p_2^{b_2} \cdots p_n^{b_n}$$

where each exponent is a nonnegative integer (possibly zero)

$$\gcd(a, b) = p_1^{\min(a_1,b_1)} p_2^{\min(a_2,b_2)} \cdots p_n^{\min(a_n,b_n)}$$

This number clearly divides $a$ and $b$. No larger number can divide both $a$ and $b$. Proof by contradiction and the prime factorisation of a postulated larger divisor.
Gcd by prime factorisations

Suppose that the prime factorisations of $a$ and $b$ are

$$a = p_1^{a_1} p_2^{a_2} \cdots p_n^{a_n}$$
$$b = p_1^{b_1} p_2^{b_2} \cdots p_n^{b_n}$$

where each exponent is a nonnegative integer (possibly zero)

$$\text{gcd}(a, b) = p_1^{\min(a_1, b_1)} p_2^{\min(a_2, b_2)} \cdots p_n^{\min(a_n, b_n)}$$

This number clearly divides $a$ and $b$. No larger number can divide both $a$ and $b$. Proof by contradiction and the prime factorisation of a postulated larger divisor.

Factorisation is a very inefficient method to compute gcd
Euclidian algorithm: efficient for computing gcd

Euclidian algorithm

algorithm gcd(x, y)
    if y = 0
    then return(x)
    else return(gcd(y, x mod y))
Euclidian algorithm: efficient for computing gcd

Euclidian algorithm

```plaintext
algorithm gcd(x, y)
    if y = 0
      then return(x)
    else return(gcd(y, x mod y))
```

The Euclidian algorithm relies on

\[ \forall x, y \in \mathbb{Z}^+ \ (\text{gcd}(x, y) = \text{gcd}(y, x \mod y)) \]
Euclidian algorithm (proof of correctness)

Lemma

If $x = yq + r$, where $x$, $y$, $q$, and $r$ are positive integers, then $\gcd(x, y) = \gcd(y, r)$. (Consider $r = x \mod y$ and $q = x \div y$)
Euclidian algorithm (proof of correctness)

Lemma

If \( x = yq + r \), where \( x, y, q, \) and \( r \) are positive integers, then \( \gcd(x, y) = \gcd(y, r) \). (Consider \( r = x \mod y \) and \( q = x \div y \))

Proof.

(\( \Rightarrow \)) Suppose that \( d \) divides both \( x \) and \( y \). Then \( d \) also divides \( x - yq = r \). Hence, any common divisor of \( x \) and \( y \) must also be a common divisor of \( y \) and \( r \).

(\( \Leftarrow \)) Suppose that \( d \) divides both \( y \) and \( r \). Then \( d \) also divides \( yq + r = x \). Hence, any common divisor of \( y \) and \( r \) must also be a common divisor of \( x \) and \( y \).

Therefore, \( \gcd(x, y) = \gcd(y, r) \)
Gcd as a linear combination

**Theorem (Bézout’s theorem)**

*If x and y are positive integers, then there exist integers a and b such that \( \gcd(x, y) = ax + by \)*
Theorem (Bézout’s theorem)

If $x$ and $y$ are positive integers, then there exist integers $a$ and $b$ such that $\text{gcd}(x, y) = ax + by$

Proof.

Nonconstructive proof. Let $S$ be the set of positive integers $ax + by$ (where $a$ or $b$ may be negative integers); $S$ is non-empty as it includes $x + y$. By the well-ordering principle $S$ has a least element $c$. So $c = ax + by$ for some $a$ and $b$. If $d|x$ and $d|y$ then $d|ax$ and $d|by$ and so $d|(ax + by)$, that is $d|c$. We now show $c|x$ and $c|y$ which means that $c = \text{gcd}(x, y)$. Assume $c \nmid x$. So $x = qc + r$ where $0 < r < c$. Now $r = x - qc = x - q(ax + by)$. That is, $r = (1 - qa)x + (-qb)y$, so $r \in S$ which contradicts that $c$ is the least element in $S$ as $r < c$. The same argument shows $c|y$. 

\[ \square \]
Bézout’s theorem: constructive proof

Extended Euclidian algorithm

algorithm e-gcd(x,y)

if y = 0
then return(x, 1, 0)
else
(d, a, b) := e-gcd(y, x mod y)
return((d, b, a - ((x div y) * b)))

e-gcd

(24, 9)
e-gcd

(22, 9)
Bézout’s theorem: constructive proof

Extended Euclidian algorithm

algorithm e-gcd(x,y)
    if y = 0
    then return(x, 1, 0)
    else
        (d, a, b) := e-gcd(y, x mod y)
    return((d, b, a - ((x div y) * b)))

e-gcd(24, 9)
e-gcd(22, 9)
Bézout’s theorem: constructive proof

Extended Euclidian algorithm

```plaintext
algorithm e-gcd(x, y)
  if y = 0
  then return(x, 1, 0)
  else
    (d, a, b) := e-gcd(y, x mod y)
  return((d, b, a - ((x div y) * b)))
```

e-gcd(24, 9)
Bézout’s theorem: constructive proof

Extended Euclidean algorithm

```
algorithm e-gcd(x, y)
  if y = 0
  then return(x, 1, 0)
  else
    (d, a, b) := e-gcd(y, x mod y)
  return((d, b, a - ((x div y) * b)))
```

- e-gcd(24, 9)
- e-gcd(22, 9)
Bézout’s theorem: constructive proof

Extended Euclidian algorithm

algorithm e-gcd(x,y)
if y = 0
then return(x, 1, 0)
else
(d, a, b) := e-gcd(y, x mod y)
return((d, b, a - ((x div y) * b)))

Correctness proof for computing Bézout coefficients

Let

\[ x = yq + r \]

where

\[ r = x \mod y \]

and

\[ q = x \div y \]

So

\[ r = x - yq \]

If \( d = ay + br \) then

\[ d = ay + b(x - yq) = bx + (a - qb)y \]

Base case

\( y = 0 \):

\[ e-gcd(x, y) = (x, 1, 0) \]

and

\[ x = 1 \times x + 0 \times y \]
Bézout’s theorem: constructive proof

Extended Euclidian algorithm

```plaintext
algorithm e-gcd(x, y)
    if y = 0
    then return(x, 1, 0)
    else
        (d, a, b) := e-gcd(y, x mod y)
        return((d, b, a - ((x div y) * b))
```
Bézout’s theorem: constructive proof

Extended Euclidian algorithm

```plaintext
algorithm e-gcd(x,y)
    if y = 0
        then return(x, 1, 0)
    else
        (d, a, b) := e-gcd(y, x mod y)
        return((d, b, a - ((x div y) * b)))
```

Correctness proof for computing Bézout coefficients

- Let \( x = yq + r \) where \( r = x \mod y \) and \( q = x \div y \)
- So \( r = x - yq \)
Bézout’s theorem: constructive proof

Extended Euclidian algorithm

```
algorithm e-gcd(x,y)
    if y = 0
        then return(x, 1, 0)
    else
        (d, a, b) := e-gcd(y, x mod y)
        return((d, b, a - ((x div y) * b)))
```

Correctness proof for computing Bézout coefficients

- Let \( x = yq + r \) where \( r = x \mod y \) and \( q = x \div y \)
- So \( r = x - yq \)
- If \( d = ay + br \) then
Bézout’s theorem: constructive proof

Extended Euclidian algorithm

```plaintext
algorithm e-gcd(x, y)
    if y = 0
        then return(x, 1, 0)
    else
        (d, a, b) := e-gcd(y, x mod y)
        return((d, b, a - ((x div y) * b)))
```

Correctness proof for computing Bézout coefficients

- Let \( x = yq + r \) where \( r = x \mod y \) and \( q = x \div y \)
- So \( r = x - yq \)
- If \( d = ay + br \) then
  \[
  d = ay + b(x - yq) = bx + (a - qb)y
  \]
Bézout’s theorem: constructive proof

Extended Euclidian algorithm

algorithm e-gcd(x,y)
    if y = 0
    then return(x, 1, 0)
    else
        (d, a, b) := e-gcd(y, x mod y)
        return((d, b, a - ((x div y) * b)))

Correctness proof for computing Bézout coefficients

- Let \( x = yq + r \) where \( r = x \mod y \) and \( q = x \div y \)
- So \( r = x - yq \)
- If \( d = ay + br \) then
  \[ d = ay + b(x - yq) = bx + (a - qb)y \]
- Base case \( y = 0 \): e-gcd(x, y) = (x, 1, 0) and \( x = 1 \times x + 0 \times y \)
Further properties

**Theorem**

If $a$, $b$, $c$ are positive integers such that $\gcd(a, b) = 1$ and $a \mid bc$ then $a \mid c$

**Proof.**

Because $\gcd(a, b) = 1$, by Bézout's theorem there are integers $s$ and $t$ such that $sa + tb = 1$. So, $sac + tbc = c$. Assume $a \mid bc$. Therefore, $a \mid tbc$ and $a \mid sac$, so $a \mid (sac + tbc)$; that is, $a \mid c$.

**Theorem**

Let $m$ be a positive integer and let $a$, $b$, $c$ be integers. If $ac \equiv bc \pmod{m}$ and $\gcd(c, m) = 1$ then $a \equiv b \pmod{m}$

**Proof.**

Because $ac \equiv bc \pmod{m}$, it follows $m \mid (ac - bc)$; so, $m \mid c(a - b)$. By the result above because $\gcd(c, m) = 1$, it follows that $m \mid (a - b)$. 
Further properties

Theorem

If \(a, b, c\) are positive integers such that \(\gcd(a, b) = 1\) and \(a|bc\) then \(a|c\).

Proof.

Because \(\gcd(a, b) = 1\), by Bézout’s theorem there are integers \(s\) and \(t\) such that \(sa + tb = 1\). So, \(sac + tbc = c\). Assume \(a|bc\). Therefore, \(a|tbc\) and \(a|sac\), so \(a|(sac + tbc)\); that is, \(a|c\). \(\square\)
Further properties

Theorem

If $a, b, c$ are positive integers such that $\gcd(a, b) = 1$ and $a | bc$ then $a | c$

Proof.

Because $\gcd(a, b) = 1$, by Bézout’s theorem there are integers $s$ and $t$ such that $sa + tb = 1$. So, $sac + tbc = c$. Assume $a | bc$. Therefore, $a | tbc$ and $a | sac$, so $a | (sac + tbc)$; that is, $a | c$.

Theorem

Let $m$ be a positive integer and let $a, b, c$ be integers. If $ac \equiv bc \pmod{m}$ and $\gcd(c, m) = 1$ then $a \equiv b \pmod{m}$
Further properties

Theorem

If $a, b, c$ are positive integers such that $\gcd(a, b) = 1$ and $a \mid bc$ then $a \mid c$

Proof.

Because $\gcd(a, b) = 1$, by Bézout’s theorem there are integers $s$ and $t$ such that $sa + tb = 1$. So, $sac + tbc = c$. Assume $a \mid bc$. Therefore, $a \mid tbc$ and $a \mid sac$, so $a \mid (sac + tbc)$; that is, $a \mid c$.

Theorem

Let $m$ be a positive integer and let $a, b, c$ be integers. If $ac \equiv bc \pmod{m}$ and $\gcd(c, m) = 1$ then $a \equiv b \pmod{m}$

Proof.

Because $ac \equiv bc \pmod{m}$, it follows $m \mid (ac - bc)$; so, $m \mid c(a - b)$. By the result above because $\gcd(c, m) = 1$, it follows that $m \mid (a - b)$. 

Colin Stirling (Informatics) Discrete Mathematics (Chap 4) Today 9 / 9