Theorem: Euclidean Algorithm

Theorem: Euclidean Algorithm

The Euclidean algorithm provides an efficient method for computing the greatest common divisor of two integers.

Statement

Let $a, b \in \mathbb{Z}$ with $b \neq 0$. Then:

$$\gcd(a, b) = \gcd(b, a \bmod b)$$

where $a \bmod b$ is the remainder when a is divided by b.

Algorithm

Given integers a and b with $a \geq b > 0$:

1. If $b = 0$, then $\gcd(a, b) = a$
2. Otherwise, compute $r = a \bmod b$
3. Replace (a, b) with (b, r) and repeat

The algorithm terminates when the remainder becomes 0.

Proof of Correctness

We need to show that $\gcd(a, b) = \gcd(b, a \bmod b)$.

Let $r = a \bmod b$. Then $a = qb + r$ for some integer q.

Step 1: Show that any common divisor of a and b is also a common divisor of b and r.

If $d \mid a$ and $d \mid b$, then $d \mid (a - qb) = r$. Thus d divides both b and r.

Step 2: Show that any common divisor of b and r is also a common divisor of a and b.

If $d \mid b$ and $d \mid r$, then $d \mid (qb + r) = a$. Thus d divides both a and b.

Therefore, the set of common divisors of (a, b) equals the set of common divisors of (b, r), so their greatest elements are equal.

Example

Find $\gcd(48, 18)$:

1. $48 = 2 \cdot 18 + 12$, so $\gcd(48, 18) = \gcd(18, 12)$
2. $18 = 1 \cdot 12 + 6$, so $\gcd(18, 12) = \gcd(12, 6)$
3. $12 = 2 \cdot 6 + 0$, so $\gcd(12, 6) = 6$

Therefore, $\gcd(48, 18) = 6$.

Extended Euclidean Algorithm

The algorithm can be extended to find integers x, y such that:

$$\gcd(a, b) = ax + by$$

This proves Bézout's identity constructively.

Time Complexity

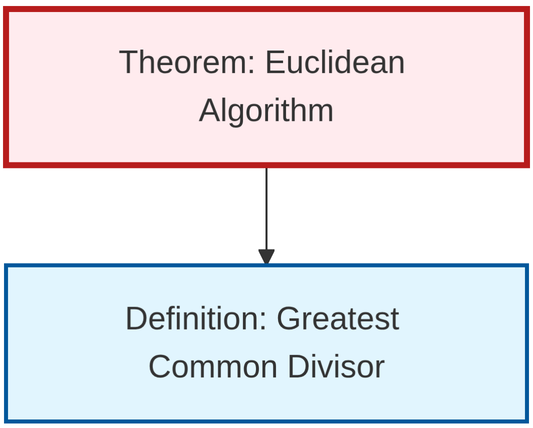
The number of steps is at most $O(\log \min(a, b))$, making it very efficient even for large numbers.

Mermaid Diagram

```
graph TD
    A[Euclidean Algorithm] --> B[gcd(a,b) = gcd(b, a mod b)]
    B --> C[Repeat until remainder = 0]
    A --> D[Example: gcd(48,18)]
    D --> E[48 = 2·18 + 12]
    E --> F[18 = 1·12 + 6]
    F --> G[12 = 2·6 + 0]
    G --> H[gcd = 6]
    A --> I[Extended Algorithm]
    I --> J[Find x,y: gcd = ax + by]

    style A fill:#f9f,stroke:#333,stroke-width:2px
    style B fill:#bfb,stroke:#333,stroke-width:2px
    style H fill:#bfb,stroke:#333,stroke-width:2px
    style J fill:#bbf,stroke:#333,stroke-width:2px
```

Dependency Graph



Local dependency graph