2 - Divisibility, Euclidean algorithm

An **integer** is a number of the form ..., -3, -2, -1, 0, 1, 2, 3, 4, ...

Division algorithm. If a and b are integers, then there exist integers q and r such that $b = qa + r, \quad 0 \leq r < a.$

Definitions.
If $r = 0$ in the division algorithm, we say that a **divides** b and write $a \mid b$.
A natural number n is **prime** if $n = ab$ implies $a = 1$ or $b = 1$.

Examples 1. $3 \mid 9$ since $9 = 3 \times 3, \quad 2 \nmid 9, \quad 9$ is not prime, $\quad 5$ is prime.

Naive algorithm. Check $2 \overset{?}{\mid} n,\ 3 \overset{?}{\mid} n, \ldots, n-1 \overset{?}{\mid} n.$

If $a \nmid n$ for all $1 < a < n$, then n is prime.

We can do better: for a composite number $n = ab$, one factor, say a, is smaller:

$$n = ab \quad \geq \quad aa = a^2 \xrightarrow{\sqrt{\ }} \sqrt{n} \geq a$$

Faster algorithm. If $a \nmid n$ for all $1 < a \leq \sqrt{n}$, then n is prime.

Example 2. Is 89 prime? Check for primes $\leq \sqrt{89} < \sqrt{100} = 10: \quad 2, 3, 5, 7.$

Contrapositive of the above statement is also true:

If n is not prime, then $a \mid n$ for some a such that $1 < a \leq \sqrt{n}$

Fundamental theorem of arithmetic. Every natural number is a unique product of increasing primes $p_1 < p_2 < p_3 < \cdots < p_r$ with some natural number multiplicities $n_1, n_2, n_3, \ldots, n_r$:

$$n = p_1^{n_1} p_2^{n_2} p_3^{n_3} \ldots p_r^{n_3}.$$

This theorem is proved with a helper theorem called a **lemma**; we omit both proofs.

Euclid's lemma. If p is a prime number and $p \mid ab$, then $p \mid a$ or $p \mid b$.

Definition. The **greatest common divisor** of two natural numbers a and b , written as $\gcd(a, b)$, is the greatest integer d such that $d \mid a$ and $d \mid b$.

Examples 3. $\gcd(17, 19) = 1$ since 17 and 19 share no common prime factors.

$$\gcd(18, 120) = \gcd(2^1 \cdot 3^2, 2^3 \cdot 3^1 \cdot 5^1) = 2^1 \cdot 3^1 = 6$$

Exercise. 1) $\gcd(210, 45) = ?$ 2) $\gcd(F0_{16}, CC_{16}) = ?$

Euclidean algorithm

Societal contexts: The hardness of factoring large integers is the basis for **RSA cryptography** and is an open problem in computer science. On quantum computers, factorization is solvable in polynomial time via **Schor's algorithm**. Due to the hardness of factoring, classical computers find gcd without factoring:

Lemma. If a divides b and c, then a divides b-c and b+c too.

Why?

$a|b$ and $a|c$ $\Rightarrow$ $b=ab'$ and $c=ac'$ for some integers b', c' $\Rightarrow$ $b\pm c=ab'\pm ac'=a(b'\pm c')$ $\Rightarrow$ $a\ |(b\pm c)$

Euclidean algorithm. If $b\div a$ gives remainder r, then $\gcd(b,a)=\gcd(a,r)$.

Why? Say $b = qa+r$. Since $\gcd(a,b)$ divides both b and qa, we know $\gcd(a,b) \mid b-qa=r$ by Lemma. Since $\gcd(a,b)$ divides a and r, we know $\gcd(a,b) \leq \gcd(a,r)$ by the definition of $\gcd(a,r)$.

Likewise, since $\gcd(a,r)$ divides both qa and r, we know $\gcd(a,r) \mid qa+r=b$ by Lemma. Since $\gcd(a,r)$ divides a and b, we know $\gcd(a,r) \leq \gcd(a,b)$ by the definition of $\gcd(a,b)$.

Example 4. Find gcd(196, 42) by the Euclidean algorithm.

$196 = 4 \times 42 + 28 \rightarrow \gcd(196, 42) = \gcd(42, 28)$

$42 = 1 \times 28 + 14 \rightarrow \quad = \gcd(28, 14)$

$28 = 2 \times 14 + 0 \rightarrow \quad = \gcd(14, 0) = 14$

The reversed Euclidean algorithm helps write gcd(a,b) as an integer combination of a and b:

Example 5. Find integers m and n such that gcd(196, 42) = 196m + 42n.

Exercise.
3A) Use the Euclidean algorithm to find gcd(170051,170).
3B) Find integers m and n such that gcd(170051,170) = 170051m + 170n.