4.7 Euclidean algorithm

**Greatest common divisor of two integers** \( m \) **and** \( n \)
is the largest integer \( d \) such that \( m = dq_1 \) and \( n = dq_2 \).

One way of finding the greatest common divisor uses the prime factorizations:
Example: \( 84 = 2^2 \cdot 3 \cdot 7 \) and \( 60 = 2^2 \cdot 3 \cdot 5 \). Clearly the \( \gcd(84, 60) = 2^2 \cdot 3 = 12 \)
For large numbers this approach may be difficult.

The Euclidean Algorithm uses the fact that the greatest common divisor of two integers must be a factor of the *difference* of the integers. If \( m = nq + r \), then the \( \gcd \) must also divide the remainder \( r \).

**Example** We will see by the Euclidean Algorithm that \( \gcd(84, 60) = 12 \):
\[
84 - 60 \times (1) = 24 \\
60 - 24 \times (2) = 12 \\
24 - 12 \times (2) = 0, \text{ so } \gcd(84, 60) = \gcd(60, 24) = \gcd(24, 12) = \gcd(12, 0) = 12
\]

**Greatest common divisor as a linear combination**

**Theorem** If \( a \) and \( b \) are positive integers and \( \gcd(a, b) = d \) then there are integers \( s \) and \( t \) such that
\[
d = s \times a + t \times b.
\]

We illustrate first a method of finding these multipliers \( s \) and \( t \) by reversing the calculations of the Euclidean Algorithm. Later we show a *direct way* of finding \( s \) and \( t \) using the **Extended Euclidean Algorithm**.

Rewriting these equations in reverse:
\[
12 = 60 + 24 \times (-2) \\
24 = 84 + 60 \times (-1)
\]
Replacing 24 in the first equation yields
\[
12 = 60 + [84 + 60 \times (-1)] \times (-2) = 84 \times (-2) + 60 \times (3).
\]

**Example** We see by Euclidean Algorithm that \( \gcd (216, 126) = 18 \)
\[
216 - 126 \times (1) = 90 \\
126 - 90 \times (1) = 36 \\
90 - 36 \times (2) = 18 \\
36 - 18 \times (2) = 0, \text{ so } \gcd(216, 126) = \gcd(126, 90) = \gcd(90, 36) = \gcd(36, 18) = 18
\]

Rewriting these equations in reverse:
\[
18 = 90 + 36 \times (-2) \text{ and } 36 = [126 + 90 \times (-1)] \text{ and } 90 = [216 + 126 \times (-1)] \text{ and substituting yields:}
\[
18 = 90 + [126 + 90 \times (-1)] \times (-2) = 126 \times (-2) + 90 \times (3)
\]
\[
= 126 \times (-2) + [216 + 126 \times (-1)] \times (3) = 216 \times (3) + 126 \times (-5).
\]

The **Extended Euclidean Algorithm** can be used to find the \( \gcd \) of two numbers and express it as a linear combination of those numbers. It uses auxiliary numbers 1 and 0 and two starting conditions to produce an invariant expression \( G = S \times A + T \times B \) that yields the desired result.

As shown in the following example, this invariant starts with \( G = 1 \times A + 0 \times B \) which equals \( A \) and continues with \( G = 0 \times A + 1 \times B \) which equals \( B \).

Initially \( A \) is integrally divided by \( B \), and the quotient \( Q \) is used to obtain the remainder \( R = A - B \times Q \).
In the example, \( Q_n = \text{floor} \left(R_{(n-2)} / R_{(n-1)} \right) \) and \( A = R_{(n-1)} \) and \( B = R_{(0)} \).
New values of \( S \) and \( T \) and \( G \) are obtained in the same way as \( R \); for example, \( S_n = S_{(n-2)} - S_{(n-1)} \times Q_n \).
The process continues until the last nonzero remainder is found; and that is the \( \gcd(A, B) \).
**Example** We can show that \( \gcd(356, 252) = 4 \) and that \( 4 = (17)356 + (-24)252 \).

In the following tableau, the first two lines express \( A \) and \( B \) as linear combinations of themselves. The calculation begins in the third line where \( Q_n = \text{floor} \left( \frac{R_{(n-2)}}{R_{(n-1)}} \right) \) and \( A = R_{(-1)} \) and \( B = R_{(0)} \). Each of the other columns uses \( Q_n \) to find the subsequent entry, and the process is repeated for each line. Specifically, \( R_n = R_{(n-2)} - R_{(n-1)} * Q_n \), and \( S_n = S_{(n-2)} - S_{(n-1)} * Q_n \), and \( T_n = T_{(n-2)} - T_{(n-1)} * Q_n \).

<table>
<thead>
<tr>
<th>( Q )</th>
<th>( R )</th>
<th>( S )</th>
<th>( T )</th>
<th>( G = SA + TB )</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>A</strong></td>
<td>356</td>
<td>1</td>
<td>0</td>
<td>356 = 1A + 0B</td>
</tr>
<tr>
<td><strong>B</strong></td>
<td>252</td>
<td>0</td>
<td>1</td>
<td>252 = 0A + 1B</td>
</tr>
<tr>
<td>1</td>
<td>104</td>
<td>1</td>
<td>-1</td>
<td>104 = 1A + (-1)B</td>
</tr>
<tr>
<td>2</td>
<td>44</td>
<td>-2</td>
<td>3</td>
<td>44 = (-2)A + 3B</td>
</tr>
<tr>
<td>2</td>
<td>16</td>
<td>5</td>
<td>-7</td>
<td>16 = 5A + (-7)B</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
<td>-12</td>
<td>17</td>
<td>12 = (-12)A + 17B</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>17</td>
<td>-24</td>
<td>4 = 17A + (-24)B</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>-63</td>
<td>89</td>
<td></td>
</tr>
</tbody>
</table>

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

Proof: Set \( 1 = sa + tb \) and multiply by \( c \), getting \( c = sac + tbc \).

Since \( a \) divides any multiple of itself and any multiple of \( bc \), \( a \mid sac \) and \( a \mid tbc \), so it must divide \( c \).

**Lemma** A prime that divides a product of integers must divide at least one of the factors.

From this lemma follows the unique factorization of positive integers.

Another consequence of the lemma is the

**Theorem** If \( m \) is a positive integer and \( a, b, c \) are integers such that \( ac \equiv bc \pmod m \) and \( \gcd(c, m) = 1 \), then \( a \equiv b \pmod m \).

**Linear Congruences** \( ax \equiv b \pmod m \)

Note that we **may not divide** in a modular system, but we can **multiply to produce one** if we know the modular inverse. It is possible to find the modular inverse by repeated trials, but the Extended Euclidean Algorithm may be more efficient when the modulus is large.

To find the modular inverse of \( 2 \pmod 7 \), consider that \( \gcd(7, 2) = 1 = (1)7 + (-3)2 \). Reducing this equation \( \pmod 7 \), where 0 replaces 7, and 4 replaces \(-3\) produces \( 1 = 0 + (4)(2) \pmod 7 \). Now to solve **the linear congruence** \( 2x \equiv 5 \pmod 7 \) we multiply by 4, which is the modular inverse of 2:

\[ 8x \equiv 20 \pmod 7 \]

which reduces to \( x \equiv 6 \pmod 7 \).

To find the modular inverse of \( 3 \pmod 13 \), consider that \( \gcd(13, 3) = 1 = (1)13 + (-4)3 \). Reducing this equation \( \pmod 13 \), where 0 replaces 13, and 9 replaces \(-4\) produces \( 1 = 0 + (9)(3) \pmod 13 \). Now to solve **the linear congruence** \( 3x \equiv 7 \pmod 13 \) we multiply by 9, which is the modular inverse of 3:

\[ 27x \equiv 63 \pmod 13 \]

which reduces to \( x \equiv 11 \pmod 13 \).

To find the modular inverse of \( 21 \pmod 26 \), consider that \( \gcd(26, 21) = 1 = (-4)26 + (5)21 \). Reducing the equation \( \pmod 26 \), where 0 replaces 26, and 5 stands for itself produces \( 1 = 0 + (5)(21) \pmod 26 \). Now to solve **the linear congruence** \( 21x \equiv 3 \pmod 26 \) we multiply by 5, which is the modular inverse of 21:

\[ 105x \equiv 15 \pmod 26 \]

which reduces to \( x \equiv 15 \pmod 26 \).