The Division Algorithm

**Task:** Compute the quotient and the remainder of two integers.

**Given:** Two positive integers \( a \) amd \( m \).

**Find:** Two integers \( q \) and \( r \) such that

\[
(1) \quad a = qm + r \quad \text{and} \quad 0 \leq r < m.
\]

**Method:** (Binary) long division.

**Note:** The integers \( q \) and \( r \) are uniquely determined by (1) and are called the *quotient* and *remainder* of division of \( a \) by \( m \), respectively. We write:

\[
\text{quo}(a, m) := q \quad \text{and} \quad \text{rem}(a, m) := r.
\]

**Analysis:** Suppose \( a \) and \( m \) have \( k \) and \( l \) bits, respectively, and that \( k \geq l \). Then:

\[
\#\text{Substractions} = \#\text{bits}(q) \leq k - l + 1 \leq k
\]

\[
\text{Time(Substraction)} = l.
\]

so the time to compute a remainder/quotient is

\[
\text{Time(rem}(a, m)) = (k - l + 1)l \leq kl
\]

\[
\text{Time(quo}(a, m)) = (k - l + 1)l \leq kl
\]
**Vista:** Much of PK Cryptography is based on *modular arithmetic*, i.e., on the Calculus of remainders. Thus, the efficient computation of remainders is very important for us.

**Application:** Computing the representation of \( n \) to the base \( b \).
- This can be done a sequence of divisions by \( b \).

**Special Case:** Computing the binary expansion of \( n \).

**Method:** Divide successive quotients by \( 2 \), list the remainders in reverse order.
- More precisely: put \( q_0 = n \) and compute successively

\[
q_{i+1} = \text{quo}(q_i, 2) \quad \text{and} \quad r_{i+1} = \text{rem}(q_i, 2),
\]

until \( q_k = 0 \). Then \((r_k, \ldots, r_1)_2\) is the binary expansion of \( n \).

**Example:** \( n = 25 \)

<table>
<thead>
<tr>
<th>( i )</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>( q_i )</td>
<td>25</td>
<td>12</td>
<td>6</td>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>( r_i )</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Thus, \( 25 = (1, 1, 0, 0, 1)_2 \) or \( 11001 \).