Lecture 1: Basic Algebraic Primitives

Rafael Oliveira

University of Waterloo
Cheriton School of Computer Science
rafael.oliveira.teaching@gmail.com

January 11, 2021
Overview

- Algebraic Primitives
- Basic Algebraic Operations
- Greatest Common Divisor
- Conclusion
- Acknowledgements
Groups

- **Group**: set $G$ with law of composition $\circ : G \times G \rightarrow G$ such that
  1. **associative**: $(a \circ b) \circ c = a \circ (b \circ c)$
  2. **identity element**: $1 \in G$ such that $1 \circ a = a \circ 1 = a$, for all $a \in G$
  3. **inverse**: every element $a \in G$ has an inverse $a^{-1} \in G$ such that

\[
a \circ a^{-1} = a^{-1} \circ a = 1
\]
Groups

- **Group**: set $G$ with law of composition $\circ : G \times G \rightarrow G$ such that
  1. **associative**: $(a \circ b) \circ c = a \circ (b \circ c)$
  2. **identity element**: $1 \in G$ such that $1 \circ a = a \circ 1 = a$, for all $a \in G$
  3. **inverse**: every element $a \in G$ has an inverse $a^{-1} \in G$ such that
     
     $$a \circ a^{-1} = a^{-1} \circ a = 1$$

- **Examples**:
  - **Invertible matrices** (quintessential example) with matrix multiplication
  - **Permutations of a set** with function composition
Groups

- **Group**: set $G$ with law of composition $\circ : G \times G \to G$ such that
  1. **associative**: $(a \circ b) \circ c = a \circ (b \circ c)$
  2. **identity element**: $1 \in G$ such that $1 \circ a = a \circ 1 = a$, for all $a \in G$
  3. **inverse**: every element $a \in G$ has an inverse $a^{-1} \in G$ such that
     \[ a \circ a^{-1} = a^{-1} \circ a = 1 \]

- **Examples**: 
  - **Invertible matrices** (quintessential example) with matrix multiplication
  - **Permutations of a set** with function composition

- $G$ is **abelian group** if the law of composition is **commutative**
  \[ a \circ b = b \circ a, \quad \forall a, b \in G \]
Groups

- **Group**: set $G$ with law of composition $\circ : G \times G \to G$ such that
  1. **associative**: $(a \circ b) \circ c = a \circ (b \circ c)$
  2. **identity element**: $1 \in G$ such that $1 \circ a = a \circ 1 = a$, for all $a \in G$
  3. **inverse**: every element $a \in G$ has an inverse $a^{-1} \in G$ such that
     \[
     a \circ a^{-1} = a^{-1} \circ a = 1
     \]

- **Examples**:
  - *Invertible matrices* (quintessential example) with *matrix multiplication*
  - *Permutations of a set* with *function composition*

- $G$ is **abelian group** if the law of composition is **commutative**
  \[
  a \circ b = b \circ a, \quad \forall a, b \in G
  \]

- **Examples of abelian groups**
  - Integers, with addition operation
  - Real numbers, with addition operation
  - Integer matrices, with addition operation
Rings\(^1\)

- **Ring**: set \( R \) with laws of composition
  - Addition \( + : R \times R \to R \)
  - Multiplication \( \cdot : R \times R \to R \)

\(^1\)Commutative rings with unit
Rings

- **Ring**: set $R$ with laws of composition
  - Addition $+: R \times R \to R$
  - Multiplication $\cdot: R \times R \to R$
- $R$ is **abelian group** with respect to addition
  - $0 \in R$ identity w.r.t. addition

---

$^1$Commutative rings with unit
**Rings**

- **Ring**: set $R$ with laws of composition
  - Addition $+ : R \times R \to R$
  - Multiplication $\cdot : R \times R \to R$
- $R$ is *abelian group* with respect to addition
  - $0 \in R$ identity w.r.t. addition
- Multiplication satisfies following properties
  - *associative*: $a \cdot (b \cdot c) = (a \cdot b) \cdot c$
  - *commutative*: $a \cdot b = b \cdot a$
  - *identity*: $1 \in R$ such that $1 \cdot a = a \cdot 1 = a$
  - *distributive over addition*:
    
    $$a \cdot (b + c) = a \cdot b + a \cdot c \quad \text{and} \quad (a + b) \cdot c = a \cdot c + b \cdot c$$

---

1 Commutative rings with unit
Rings

- **Ring**: set $R$ with laws of composition
  - Addition $+: R \times R \to R$
  - Multiplication $\cdot: R \times R \to R$
- $R$ is *abelian group* with respect to addition
  - $0 \in R$ identity w.r.t. addition
- Multiplication satisfies following properties
  - *associative*: $a \cdot (b \cdot c) = (a \cdot b) \cdot c$
  - *commutative*: $a \cdot b = b \cdot a$
  - *identity*: $1 \in R$ such that $1 \cdot a = a \cdot 1 = a$
  - *distributive over addition*:
    
    $$a \cdot (b + c) = a \cdot b + a \cdot c \quad \text{and} \quad (a + b) \cdot c = a \cdot c + b \cdot c$$
- **Examples**
  - Integers with addition and multiplication (quintessential example)
  - Real numbers, complex numbers, with usual addition and multiplication
  - Polynomial rings (quintessential example)

---

1. Commutative rings with unit
Rings - Definitions

- **Unit**: an element \( u \in R \) is a unit if there is \( v \in R \) such that \( uv = 1 \)
Rings - Definitions

- **Unit**: an element $u \in R$ is a unit if there is $v \in R$ such that $uv = 1$
- **Associates**: two elements $a, b \in R$ are associates if there is a unit $u \in R$ such that $a = ub$

\[ \mathbb{Z} \text{ units: } \{ -1, 1 \} \]

\[ 3, -3 \quad a, -a \]
Rings - Definitions

- **Unit:** an element \( u \in R \) is a unit if there is \( v \in R \) such that \( uv = 1 \)

- **Associates:** two elements \( a, b \in R \) are associates if there is a unit \( u \in R \) such that \( a = ub \)

- **Zero divisor:** a zero divisor in \( R \) is an element \( a \in R \setminus \{0\} \) such that there is a non-zero \( b \in R \setminus \{0\} \) such that \( a \cdot b = 0 \)
Rings - Definitions

- **Unit:** an element $u \in R$ is a unit if there is $v \in R$ such that $uv = 1$
- **Associates:** two elements $a, b \in R$ are associates if there is a unit $u \in R$ such that $a = ub$
- **Zero divisor:** a zero divisor in $R$ is an element $a \in R \setminus \{0\}$ such that there is a non-zero $b \in R \setminus \{0\}$ such that $a \cdot b = 0$
- **Integral domain:** a ring $R$ is an integral domain if it has *no zero divisor.*
Rings - Definitions

- **Unit:** an element $u \in R$ is a unit if there is $v \in R$ such that $uv = 1$
- **Associates:** two elements $a, b \in R$ are associates if there is a unit $u \in R$ such that $a = ub$
- **Zero divisor:** a zero divisor in $R$ is an element $a \in R \setminus \{0\}$ such that there is a non-zero $b \in R \setminus \{0\}$ such that $a \cdot b = 0$
- **Integral domain:** a ring $R$ is an integral domain if it has no zero divisor.
- **Euclidean domain:** a ring $R$ is an Euclidean domain if:
  - $R$ is an integral domain and there is an Euclidean function $|\cdot| : R \to \mathbb{N} \cup \{-\infty\}$
  - for all $a, b \in R$, with $b \neq 0$, there exists $q, r \in R$ such that
    \[ a = qb + r \text{ and } |r| < |b| \]

$\mathbb{Q}[x, y]$ is not Euclidean domain
Rings - Definitions

- **Unit:** an element \( u \in R \) is a unit if there is \( v \in R \) such that \( uv = 1 \)

- **Associates:** two elements \( a, b \in R \) are associates if there is a unit \( u \in R \) such that \( a = ub \)

- **Zero divisor:** a zero divisor in \( R \) is an element \( a \in R \setminus \{0\} \) such that there is a non-zero \( b \in R \setminus \{0\} \) such that \( a \cdot b = 0 \)

- **Integral domain:** a ring \( R \) is an integral domain if it has **no zero divisor**.

- **Euclidean domain:** a ring \( R \) is an Euclidean domain if:
  - \( R \) is an integral domain and there is an Euclidean function \( | \cdot | : R \rightarrow \mathbb{N} \cup \{-\infty\} \)
  - for all \( a, b \in R \), with \( b \neq 0 \), there exists \( q, r \in R \) such that
    \[
    a = qb + r \quad \text{and} \quad |r| < |b|
    \]

- **Greatest common divisor:** the greatest common divisor of \( a, b \in R \), denoted by \( \gcd(a, b) \) is an element of \( R \) which divides both \( a \) and \( b \), and if \( c \in R \) divides \( a \) and \( b \), then \( c \) divides \( \gcd(a, b) \).
Fields

- **Field**: a ring $\mathbb{F}$ with addition and multiplication such that
  - every non-zero element has a multiplicative inverse
Fields

- **Field**: a ring $\mathbb{F}$ with addition and multiplication such that
  - every non-zero element has a multiplicative inverse

- **Examples**
  - Rational numbers
  - Real numbers
  - Complex numbers
  - Set of integers modulo a prime
Polynomial Rings

- Given a base ring $R$, we can construct a polynomial ring $R[x]$ by “adding a new variable” $x$ to $R$ in the *freest way possible*
Polynomial Rings

- Given a base ring \( R \), we can construct a polynomial ring \( R[x] \) by “adding a new variable” \( x \) to \( R \) in the \textit{freest way possible}.

- That is:

\[
a(x) = a_0 + a_1 x + \cdots + a_d x^d = b_0 + b_1 x + \cdots + b_e x^e, \quad (a_d, b_e \neq 0)
\]

if, and only if, \( d = e \) and \( a_0 = b_0, a_1 = b_1, \ldots, a_d = b_d \).
Polynomial Rings

- Given a base ring $R$, we can construct a polynomial ring $R[x]$ by “adding a new variable” $x$ to $R$ in the freest way possible.

- That is:

  $$a_0 + a_1x + \cdots + a_dx^d = b_0 + b_1x + \cdots + b_ex^e, \quad (a_d, b_e \neq 0)$$

  if, and only if, $d = e$ and $a_0 = b_0, a_1 = b_1, \ldots, a_d = b_d$.

- Can create the polynomial ring $R[x_1, \ldots, x_n]$ by adding the variables $x_1, \ldots, x_n$ freely as above.
Given a base ring $R$, we can construct a polynomial ring $R[x]$ by “adding a new variable” $x$ to $R$ in the *freest way possible*. That is:

$$a_0 + a_1x + \cdots + a_dx^d = b_0 + b_1x + \cdots + b_ex^e, \quad (a_d, b_e \neq 0)$$

if, and only if, $d = e$ and $a_0 = b_0, a_1 = b_1, \ldots, a_d = b_d$.

Can create the polynomial ring $R[x_1, \ldots, x_n]$ by adding the variables $x_1, \ldots, x_n$ freely as above.

What is our computational model to compute polynomials?
Given a base ring $R$, we can construct a polynomial ring $R[x]$ by "adding a new variable" $x$ to $R$ in the **freest way possible**. That is:

$$a_0 + a_1 x + \cdots + a_d x^d = b_0 + b_1 x + \cdots + b_e x^e, \quad (a_d, b_e \neq 0)$$

if, and only if, $d = e$ and $a_0 = b_0, a_1 = b_1, \ldots, a_d = b_d$

Can create the polynomial ring $R[x_1, \ldots, x_n]$ by adding the variables $x_1, \ldots, x_n$ freely as above.

What is our computational model to compute polynomials?

How can we measure computational complexity in such base rings?
Complexity measures in rings

- $\mathbb{Z} \rightarrow$ bit complexity of integer

- $\lg a := \begin{cases} 1, & \text{if } a = 0 \\ 1 + \lfloor \log |a| \rfloor, & \text{otherwise} \end{cases}$
Complexity measures in rings

- $\mathbb{Z} \rightarrow \text{bit complexity of integer}$
  - $\lg a := \begin{cases} 1, & \text{if } a = 0 \\ 1 + \lfloor \log |a| \rfloor, & \text{otherwise} \end{cases}$

- $\mathbb{Q} \rightarrow \text{complexity of } a/b \text{ is the total bit complexity of } a \text{ and } b$
Complexity measures in rings

- \( \mathbb{Z} \rightarrow \) bit complexity of integer
  \[ \lg a := \begin{cases} 1, & \text{if } a = 0 \\ 1 + \lfloor \log |a| \rfloor, & \text{otherwise} \end{cases} \]

- \( \mathbb{Q} \rightarrow \) complexity of \( a/b \) is the total bit complexity of \( a \) and \( b \)

- \( \mathbb{F}_q \rightarrow \) complexity of element is bit complexity (log \( q \))
Complexity measures in rings

- $\mathbb{Z} \rightarrow$ bit complexity of integer
  
  \[ \lg a := \begin{cases} 
  1, & \text{if } a = 0 \\
  1 + \lfloor \log |a| \rfloor, & \text{otherwise} 
  \end{cases} \]

- $\mathbb{Q} \rightarrow$ complexity of $a/b$ is the total bit complexity of $a$ and $b$

- $\mathbb{F}_q \rightarrow$ complexity of element is bit complexity ($\log q$)

- Polynomial rings $R[x_1, \ldots, x_n]$
  
  1. dense representation
Complexity measures in rings

- \( \mathbb{Z} \rightarrow \) bit complexity of integer
  
  \[
  \lg a := \begin{cases} 
  1, & \text{if } a = 0 \\
  1 + \lfloor \log |a| \rfloor, & \text{otherwise}
  \end{cases}
  \]

- \( \mathbb{Q} \rightarrow \) complexity of \( a/b \) is the total bit complexity of \( a \) and \( b \)

- \( \mathbb{F}_q \rightarrow \) complexity of element is bit complexity (\( \log q \))

- Polynomial rings \( R[x_1, \ldots, x_n] \)
  
  1. dense representation
  2. sparse representation
Complexity measures in rings

- \( \mathbb{Z} \rightarrow \) bit complexity of integer
  - \( \lg a := \begin{cases} 1, & \text{if } a = 0 \\ 1 + \lceil \log |a| \rceil, & \text{otherwise} \end{cases} \)
- \( \mathbb{Q} \rightarrow \) complexity of \( a/b \) is the total bit complexity of \( a \) and \( b \)
- \( \mathbb{F}_q \rightarrow \) complexity of element is bit complexity (\( \log q \))
- Polynomial rings \( R[x_1, \ldots, x_n] \)
  1. dense representation
  2. sparse representation
  3. algebraic circuits
• Algebraic Primitives

• Basic Algebraic Operations

• Greatest Common Divisor

• Conclusion

• Acknowledgements
Addition and Multiplication over $R = \mathbb{Z}$

- **Input:** two elements $a, b \in \mathbb{Z}$
- **Output:** $a + b$
Addition and Multiplication over $R = \mathbb{Z}$

- **Input:** two elements $a, b \in \mathbb{Z}$
- **Output:** $a + b$
- Look at the bit representation of $a, b$ – perform addition with carrying.
Addition and Multiplication over $R = \mathbb{Z}$

- **Input:** two elements $a, b \in \mathbb{Z}$
- **Output:** $a + b$
- Look at the bit representation of $a, b$ – perform addition with carrying.
- Running time: $O(\lg a + \lg b)$

$$\leq c(\lg a + \lg b)$$
Addition and Multiplication over $R = \mathbb{Z}$

- **Input:** two elements $a, b \in \mathbb{Z}$
- **Output:** $a + b$
- Look at the bit representation of $a, b$ – perform addition with carrying.
- Running time: $O(\lg a + \lg b)$

- **Input:** two elements $a, b \in \mathbb{Z}$
- **Output:** $a \cdot b$
Addition and Multiplication over $R = \mathbb{Z}$

- **Input:** two elements $a, b \in \mathbb{Z}$
- **Output:** $a + b$
- Look at the bit representation of $a, b$ – perform addition with carrying.
- Running time: $O(\lg a + \lg b)$

- **Input:** two elements $a, b \in \mathbb{Z}$
- **Output:** $a \cdot b$
- Look at bit representation of $a, b$
- Perform $\lceil \lg b \rceil$ additions of multiples of $a$
Addition and Multiplication over $R = \mathbb{Z}$

- **Input:** two elements $a, b \in \mathbb{Z}$
- **Output:** $a + b$
- Look at the bit representation of $a, b$ – perform addition with carrying.
- Running time: $O(\lg a + \lg b)$

- **Input:** two elements $a, b \in \mathbb{Z}$
- **Output:** $a \cdot b$
- Look at bit representation of $a, b$
- Perform $\lceil \lg b \rceil$ additions of multiples of $a$
- Running time: $O(\lg a \cdot \lg b)$
Naive upper bounds

<table>
<thead>
<tr>
<th>Operation</th>
<th>over ring ( \mathbb{Z} )</th>
<th>over ring ( \mathbb{Z}[x] )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( a + b )</td>
<td>( \lg(a) + \lg(b) )</td>
<td></td>
</tr>
<tr>
<td>( a \cdot b )</td>
<td>( \lg(a) \cdot \lg(b) )</td>
<td></td>
</tr>
<tr>
<td>( a = qb + r )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \gcd(a, b) )</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table:** Naive upper bounds

- over \( \mathbb{Z} \) we count word operations
- over \( \mathbb{Z}[x] \) we count operations in \( \mathbb{Z} \)
- \( \deg(a) = m, \deg(b) = n \)
Addition and multiplication over $\mathbb{Z}[x]$

- **Input:** two elements $a, b \in \mathbb{Z}[x]$, $\deg(a) = m$, $\deg(b) = n$
- **Output:** $c = a + b$
Addition and multiplication over $\mathbb{Z}[x]$

- **Input:** two elements $a, b \in \mathbb{Z}[x]$, $\deg(a) = m$, $\deg(b) = n$
- **Output:** $c = a + b$
- $c_i = a_i + b_i$ for $0 \leq i \leq \max(m, n)$
Addition and multiplication over \( \mathbb{Z}[x] \)

- **Input:** two elements \( a, b \in \mathbb{Z}[x] \), \( \deg(a) = m \), \( \deg(b) = n \)
- **Output:** \( c = a + b \)
- \( c_i = a_i + b_i \) for \( 0 \leq i \leq \max(m, n) \)
- Running time: \( O(m + n) \)
Addition and multiplication over $\mathbb{Z}[x]$

**Input:** two elements $a, b \in \mathbb{Z}[x]$, deg$(a) = m$, deg$(b) = n$

**Output:** $c = a + b$

$c_i = a_i + b_i$ for $0 \leq i \leq \max(m, n)$

Running time: $O(m + n)$

**Input:** two elements $a, b \in \mathbb{Z}[x]$

**Output:** $a \cdot b$
Addition and multiplication over \( \mathbb{Z}[x] \)

- **Input:** two elements \( a, b \in \mathbb{Z}[x] \), \( \deg(a) = m \), \( \deg(b) = n \)
- **Output:** \( c = a + b \)
- \( c_i = a_i + b_i \) for \( 0 \leq i \leq \max(m, n) \)
- Running time: \( O(m + n) \)

- **Input:** two elements \( a, b \in \mathbb{Z}[x] \)
- **Output:** \( a \cdot b \)
- \( c_k = \sum_{i=0}^{k} a_i b_{k-i} \)

\[
Q = a_0 + a_1 x + \cdots + a_m x^m \\
b = b_0 + b_1 x + \cdots + b_n x^n \\
c_0 = a_0 \cdot b_0 \\
c_1 = a_0 \cdot b_1 + a_1 b_0
\]
Addition and multiplication over $\mathbb{Z}[x]$

- **Input:** two elements $a, b \in \mathbb{Z}[x]$, $\text{deg}(a) = m$, $\text{deg}(b) = n$
- **Output:** $c = a + b$
- $c_i = a_i + b_i$ for $0 \leq i \leq \text{max}(m, n)$
- Running time: $O(m + n)$

- **Input:** two elements $a, b \in \mathbb{Z}[x]$
- **Output:** $a \cdot b$
- $c_k = \sum_{i=0}^{k} a_i b_{k-i}$
- Compute all multiplications $a_i b_j$, there are $(m + 1)(n + 1)$ of them
- Add them all properly
- Running time: $O(m \cdot n)$
Naive upper bounds

<table>
<thead>
<tr>
<th>Operation</th>
<th>over ring $\mathbb{Z}$</th>
<th>over ring $\mathbb{Z}[x]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a + b$</td>
<td>$\lg(a) + \lg(b)$</td>
<td>$m + n + 1$</td>
</tr>
<tr>
<td>$a \cdot b$</td>
<td>$\lg(a) \cdot \lg(b)$</td>
<td>$(m + 1)(n + 1)$</td>
</tr>
<tr>
<td>$a = qb + r$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\gcd(a, b)$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table: Naive upper bounds

- over $\mathbb{Z}$ we count word operations
- over $\mathbb{Z}[x]$ we count operations in $\mathbb{Z}$
- $\deg(a) = m$, $\deg(b) = n$
Division with remainder over \( \mathbb{Z} \)

**Input:** two elements \( a, b \in \mathbb{Z} \), with \( b \) non-zero

**Output:** \( q, r \in \mathbb{Z} \) such that \(|r| < |b|\) and \( a = q \cdot b + r\)
Division with remainder over \( \mathbb{Z} \)

- **Input**: two elements \( a, b \in \mathbb{Z} \), with \( b \) non-zero
- **Output**: \( q, r \in \mathbb{Z} \) such that \( |r| < |b| \) and \( a = q \cdot b + r \)
- Start with \( r = a, q = 0 \)
Division with remainder over $\mathbb{Z}$

- **Input:** two elements $a, b \in \mathbb{Z}$, with $b$ non-zero
- **Output:** $q, r \in \mathbb{Z}$ such that $|r| < |b|$ and $a = q \cdot b + r$

Start with $r = a$, $q = 0$

While $|r| \geq |b|$:  
  - $q \leftarrow q + 1$
  - $r \leftarrow r - b$

\[
\begin{align*}
 a &= 0 \cdot b + a \\
 a &= 1 \cdot b + (a-b)
\end{align*}
\]
Division with remainder over $\mathbb{Z}$

- **Input:** two elements $a, b \in \mathbb{Z}$, with $b$ non-zero
- **Output:** $q, r \in \mathbb{Z}$ such that $|r| < |b|$ and $a = q \cdot b + r$

Start with $r = a$, $q = 0$

While $|r| \geq |b|$: 
  - $q \leftarrow q + 1$
  - $r \leftarrow r - b$

Analysis: we will perform $\lfloor a/b \rfloor$ subtractions to $r$. Total time $\frac{a \lg b}{b}$
Division with remainder over \( \mathbb{Z} \)

- **Input:** two elements \( a, b \in \mathbb{Z} \), with \( b \) non-zero
- **Output:** \( q, r \in \mathbb{Z} \) such that \( |r| < |b| \) and \( a = q \cdot b + r \)
- Start with \( r = a, q = 0 \)
- While \( |r| \geq |b| \):
  - \( q \leftarrow q + 1 \)
  - \( r \leftarrow r - b \)

**Analysis:** we will perform \( \lceil a/b \rceil \) subtractions to \( r \). Total time \( \frac{a \lg b}{b} \)

- While \( |r| \geq |b| \):
  - \( q \leftarrow q + 2^{\lg r - \lg b} \)
  - \( r \leftarrow r - 2^{\lg r - \lg b} \cdot b \) \( \) kills most significant bit of \( r \)
Division with remainder over $\mathbb{Z}$

- **Input**: two elements $a, b \in \mathbb{Z}$, with $b$ non-zero
- **Output**: $q, r \in \mathbb{Z}$ such that $|r| < |b|$ and $a = q \cdot b + r$
- Start with $r = a$, $q = 0$
- While $|r| \geq |b|$: 
  - $q \leftarrow q + 1$
  - $r \leftarrow r - b$
- Analysis: we will perform $\lfloor a/b \rfloor$ subtractions to $r$. Total time $\frac{a \cdot \lg b}{b}$
- While $|r| \geq |b|$: 
  - $q \leftarrow q + 2^{\lg r - \lg b}$
  - $r \leftarrow r - 2^{\lg r - \lg b} \cdot b$
- Analysis: we will perform $\lg(a/b) = \lg(q)$ subtractions to $r$. Total time $\lg q \cdot \lg b$
Division with remainder over $\mathbb{Z}[x]$

- **Input:** two elements $a, b \in \mathbb{Z}[x]$, with $b$ non-zero and $\text{LC}(b)$ unit in $\mathbb{Z}$

- **Output:** $q, r \in \mathbb{Z}[x]$ such that $\deg(r) < \deg(b)$ and $a = q \cdot b + r$
Division with remainder over $\mathbb{Z}[x]$

- **Input:** two elements $a, b \in \mathbb{Z}[x]$, with $b$ non-zero and $LC(b)$ unit in $\mathbb{Z}$
- **Output:** $q, r \in \mathbb{Z}[x]$ such that $\deg(r) < \deg(b)$ and $a = q \cdot b + r$
- Start with $r = a$, $q = 0$
Division with remainder over \( \mathbb{Z}[x] \)

- **Input:** two elements \( a, b \in \mathbb{Z}[x] \), with \( b \) non-zero and \( LC(b) \) unit in \( \mathbb{Z} \)
- **Output:** \( q, r \in \mathbb{Z}[x] \) such that \( \deg(r) < \deg(b) \) and \( a = q \cdot b + r \)
- Start with \( r = a, q = 0 \)
- While \( \deg(r) \geq \deg(b) \):
  - \( q \leftarrow q + x^{\deg(r) - \deg(b)} \cdot \frac{LC(a)}{LC(b)} \)
  - \( r \leftarrow r - x^{\deg(r) - \deg(b)} \cdot \frac{LC(r)}{LC(b)} \cdot b \)

**killing \( \text{LT}(r) \)**

(decreasing the degree of \( r \))
Division with remainder over $\mathbb{Z}[x]$

- **Input**: two elements $a, b \in \mathbb{Z}[x]$, with $b$ non-zero and $LC(b)$ unit in $\mathbb{Z}$
- **Output**: $q, r \in \mathbb{Z}[x]$ such that $\text{deg}(r) < \text{deg}(b)$ and $a = q \cdot b + r$
- Start with $r = a$, $q = 0$
- While $\text{deg}(r) \geq \text{deg}(b)$:
  - $q \leftarrow q + x^{\text{deg}(r) - \text{deg}(b)}$
  - $r \leftarrow r - x^{\text{deg}(r) - \text{deg}(b)} \cdot \frac{LC(r)}{LC(b)} \cdot b$
- **Analysis**: we will perform at most $\text{deg}(a) - \text{deg}(b) + 1$ subtractions to $r$. Total time $(\text{deg}(a) - \text{deg}(b) + 1)(\text{deg}(b) + 1)$. 
## Naive upper bounds

<table>
<thead>
<tr>
<th>Operation</th>
<th>over ring $\mathbb{Z}$</th>
<th>over ring $\mathbb{Z}[x]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a + b$</td>
<td>$\lg(a) + \lg(b)$</td>
<td>$m + n + 1$</td>
</tr>
<tr>
<td>$a \cdot b$</td>
<td>$\lg(a) \cdot \lg(b)$</td>
<td>$(m + 1)(n + 1)$</td>
</tr>
<tr>
<td>$a = qb + r$</td>
<td>$\lg(q) \cdot \lg(b)$</td>
<td>$(n + 1)(m - n + 1)$</td>
</tr>
<tr>
<td>$\text{gcd}(a, b)$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table:** Naive upper bounds

- over $\mathbb{Z}$ we count word operations
- over $\mathbb{Z}[x]$ we count operations in $\mathbb{Z}$
- $\deg(a) = m$, $\deg(b) = n$
• Algebraic Primitives

• Basic Algebraic Operations

• Greatest Common Divisor

• Conclusion

• Acknowledgements
Extended Euclidean Algorithm

- Let $R$ be Euclidean domain, with $|\cdot|$ being its size function.
- **Input:** two elements $a, b \in R$, with $b$ non-zero
- **Output:** $s, t \in R$ such that $\text{gcd}(a, b) = as + bt$
**Extended Euclidean Algorithm**

- Let $R$ be Euclidean domain, with $|\cdot|$ being its size function.
- **Input**: two elements $a, b \in R$, with $b$ non-zero
- **Output**: $s, t \in R$ such that $\gcd(a, b) = as + bt$
- Not only do we compute the GCD, but we also express it as linear combination of $a, b$ (and we prove that such combination exists via an algorithm!)
Extended Euclidean Algorithm

- Let $R$ be Euclidean domain, with $| \cdot |$ being its size function.
- **Input:** two elements $a, b \in R$, with $b$ non-zero
- **Output:** $s, t \in R$ such that $\gcd(a, b) = as + bt$
- Not only do we compute the GCD, but we also express it as linear combination of $a, b$ (and we prove that such combination exists via an algorithm!)
- **Example:** $a = 77, b = 63$
Extended Euclidean Algorithm

- Let $R$ be Euclidean domain, with $|\cdot|$ being its size function.
- **Input:** two elements $a, b \in R$, with $b$ non-zero
- **Output:** $s, t \in R$ such that $\gcd(a, b) = as + bt$

Not only do we compute the GCD, but we also express it as linear combination of $a, b$ (and we prove that such combination exists via an algorithm!)

- Example: $a = 77, b = 63$
- $r_0 = a, r_1 = b, s = t = 0$
Extended Euclidean Algorithm

- Let $R$ be Euclidean domain, with $|\cdot|$ being its size function.
- **Input:** two elements $a, b \in R$, with $b$ non-zero
- **Output:** $s, t \in R$ such that $\gcd(a, b) = as + bt$
- Not only do we compute the GCD, but we also express it as linear combination of $a, b$ (and we prove that such combination exists via an algorithm!)
- Example: $a = 77, b = 63$
- $r_0 = a, r_1 = b, s = t = 0$
- For $1 \leq i$, let $q_i, r_{i+1}$ be such that

$$r_{i-1} = q_i r_i + r_{i+1}$$
Extended Euclidean Algorithm

- Let $R$ be Euclidean domain, with $|\cdot|$ being its size function.
- **Input:** two elements $a, b \in R$, with $b$ non-zero
- **Output:** $s, t \in R$ such that $\gcd(a, b) = as + bt$
- Not only do we compute the GCD, but we also express it as linear combination of $a, b$ (and we prove that such combination exists via an algorithm!)
- Example: $a = 77, b = 63$
- $r_0 = a, r_1 = b, s = t = 0$
- For $1 \leq i$, let $q_i, r_{i+1}$ be such that

  \[ r_{i-1} = q_i r_i + r_{i+1} \]

- while $r_i \neq 0$, continue the procedure above.
Extended Euclidean Algorithm

- Let $R$ be Euclidean domain, with $|\cdot|$ being its size function.
- **Input:** two elements $a, b \in R$, with $b$ non-zero
- **Output:** $s, t \in R$ such that $\gcd(a, b) = as + bt$

Not only do we compute the GCD, but we also express it as linear combination of $a, b$ (and we prove that such combination exists via an algorithm!)

- Example: $a = 77, b = 63$
- $r_0 = a, r_1 = b, s = t = 0$
- For $1 \leq i$, let $q_i, r_{i+1}$ be such that

  $$r_{i-1} = q_i r_i + r_{i+1}$$

- while $r_i \neq 0$, continue the procedure above.
- it will eventually stop because $|r_1| > |r_2| > \cdots$ and size function is well-ordered.
Extended Euclidean Algorithm - Correctness

- \( r_0 = a, r_1 = b, s = t = 0 \)
- For \( 1 \leq i \), let \( q_i, r_{i+1} \) be such that

\[
    r_{i-1} = q_i r_i + r_{i+1}
\]

- Suppose procedure stopped at \( r_{\ell+1} = 0 \). Show that \( r_\ell = \gcd(a, b) \).
Extended Euclidean Algorithm - Running time I

- $r_0 = a, r_1 = b, s = t = 0$
- For $1 \leq i$, let $q_i, r_{i+1}$ be such that
  \[ r_{i-1} = q_i r_i + r_{i+1} \]
- Suppose procedure stopped at $r_{\ell+1} = 0$. 
Extended Euclidean Algorithm - Running time II

- \( r_0 = a, r_1 = b, s = t = 0 \)
- For \( 1 \leq i \), let \( q_i, r_{i+1} \) be such that
  \[
  r_{i-1} = q_i r_i + r_{i+1}
  \]
- Suppose procedure stopped at \( r_{\ell+1} = 0 \).
Naive upper bounds

<table>
<thead>
<tr>
<th>Operation</th>
<th>over ring $\mathbb{Z}$</th>
<th>over ring $\mathbb{Z}[x]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a + b$</td>
<td>$\lg(a) + \lg(b)$</td>
<td>$m + n + 1$</td>
</tr>
<tr>
<td>$a \cdot b$</td>
<td>$\lg(a) \cdot \lg(b)$</td>
<td>$(m + 1)(n + 1)$</td>
</tr>
<tr>
<td>$a = qb + r$</td>
<td>$\lg(q) \cdot \lg(b)$</td>
<td>$(n + 1)(m - n + 1)$</td>
</tr>
<tr>
<td>gcd($a$, $b$)</td>
<td>$\lg(a) \cdot \lg(b)$</td>
<td>$(m + 1)(n + 1)$</td>
</tr>
</tbody>
</table>

Table: Naive upper bounds

- over $\mathbb{Z}$ we count word operations
- over $\mathbb{Z}[x]$ we count operations in $\mathbb{Z}$
- $\deg(a) = m$, $\deg(b) = n$
Acknowledgement

- Lecture based largely on:
  - Lecture 2 from CS 487 Winter 2020 - see references in suggested reading