The Division Algorithm

Michelle Hewson

September 17, 2021
Math 479 Project #1

Introduction

- Division is one of the basic mathematical operations that is used in daily life.

Introduction

- Division is one of the basic mathematical operations that is used in daily life.
- The division of integers is conceptualized by the Division Algorithm.

Introduction

- Division is one of the basic mathematical operations that is used in daily life.
- The division of integers is conceptualized by the Division Algorithm.
  - This algorithm falls under the scope of Euclidean division.

Introduction

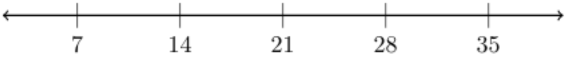
- Division is one of the basic mathematical operations that is used in daily life.
- The division of integers is conceptualized by the Division Algorithm.
  - This algorithm falls under the scope of Euclidean division.
- Its proof and applications are diverse and important in theoretical and applied mathematics.

Introduction

Consider the following example: $26 \div 7$

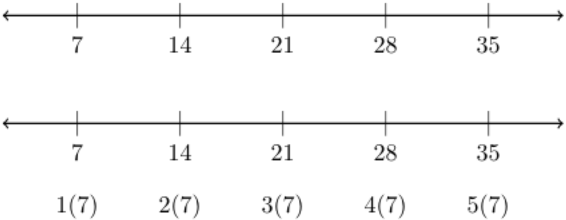
Introduction

Consider the following example: $26 \div 7$



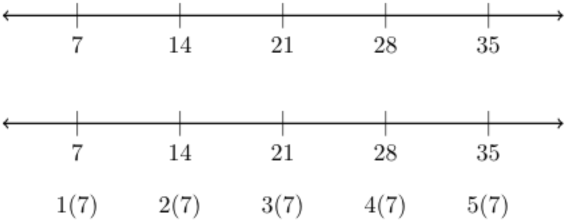
Introduction

Consider the following example: $26 \div 7$



Introduction

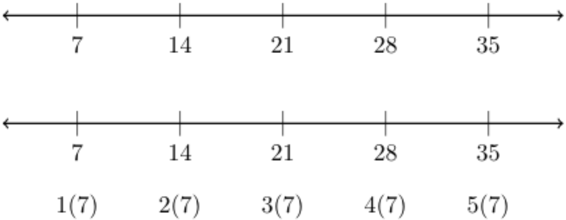
Consider the following example: $26 \div 7$



$$26 = 7 + 7 + 7 + r$$

Introduction

Consider the following example: $26 \div 7$



$$26 = 7 + 7 + 7 + r$$

$$26 = 3(7) + 5$$

Definitions

- The **quotient** is the result of dividing one integer by another.

Definitions

- The **quotient** is the result of dividing one integer by another.

$$\text{dividend} \div \text{divisor} = \text{quotient}$$

Definitions

- The **quotient** is the result of dividing one integer by another.

$$\text{dividend} \div \text{divisor} = \text{quotient}$$

- The **remainder** is the 'left-over' integer after dividing one integer by another to produce an integer quotient.

Definitions

- The **quotient** is the result of dividing one integer by another.

$$\text{dividend} \div \text{divisor} = \text{quotient}$$

- The **remainder** is the 'left-over' integer after dividing one integer by another to produce an integer quotient.

$$\text{dividend} \div \text{divisor} = \text{quotient} + \text{remainder}$$

Definitions

Theorem (Well Ordering Principle)

Every nonempty set of non-negative integers has a smallest element.

- This is strictly applicable to nonempty sets with positive elements.

Definitions

Theorem (Well Ordering Principle)

Every nonempty set of non-negative integers has a smallest element.

- This is strictly applicable to nonempty sets with positive elements.
- We will use the Well Ordering Principle to prove the Division Algorithm.

The Division Algorithm

Theorem

If a and b are integers, with $b > 0$, there exist unique integers q and r such that

$$a = qb + r \quad \textit{with} \quad 0 \leq r < b.$$

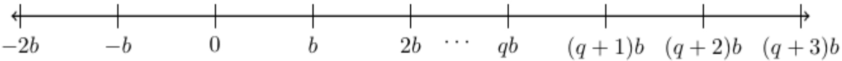
The integer q represents the quotient and r represents the remainder of the division of a by b.

The Division Algorithm Visualization

Theorem

If a and b are integers, with $b > 0$, there exist unique integers q and r such that

$$a = qb + r \quad \text{with} \quad 0 \leq r < b.$$

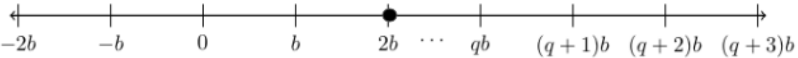


The Division Algorithm Visualization

Theorem

If a and b are integers, with $b > 0$, there exist unique integers q and r such that

$$a = qb + r \quad \textit{with} \quad 0 \le r < b.$$

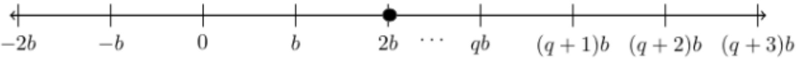


The Division Algorithm Visualization

Theorem

If a and b are integers, with $b > 0$, there exist unique integers q and r such that

$$a = qb + r \quad \textit{with} \quad 0 \leq r < b.$$



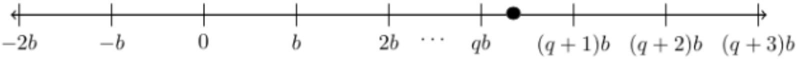
$$r = 0$$

The Division Algorithm Visualization

Theorem

If a and b are integers, with $b > 0$, there exist unique integers q and r such that

$$a = qb + r \quad \textit{with} \quad 0 \leq r < b.$$

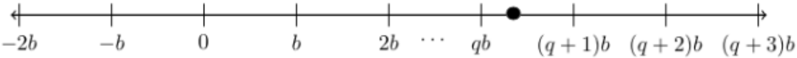


The Division Algorithm Visualization

Theorem

If a and b are integers, with $b > 0$, there exist unique integers q and r such that

$$a = qb + r \quad \text{with} \quad 0 \leq r < b.$$



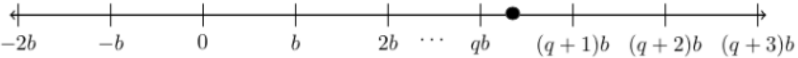
$$r \neq 0$$

The Division Algorithm Visualization

Theorem

If a and b are integers, with $b > 0$, there exist unique integers q and r such that

$$a = qb + r \quad \text{with} \quad 0 \leq r < b.$$



$$r \neq 0$$

$$0 < r < b$$

Idea of the Proof of the Division Algorithm

Idea of the Proof:

- Show that q and r both exist and are unique integer values.

Idea of the Proof of the Division Algorithm

Idea of the Proof:

- Show that q and r both **exist**.

Idea of the Proof of the Division Algorithm

Idea of the Proof:

- Show that q and r both **exist**.
  - Create a generalized set of remainders.

Idea of the Proof of the Division Algorithm

Idea of the Proof:

- Show that q and r both **exist**.
  - Create a generalized set of remainders.
  - Utilize the Well Ordering Principle to prove q and r exist in this set.

Idea of the Proof of the Division Algorithm

Idea of the Proof:

- Show that q and r both **exist**.
  - Create a generalized set of remainders.
  - Utilize the Well Ordering Principle to prove q and r exist in this set.
- Show that q and r are both **unique** integer values.

Idea of the Proof of the Division Algorithm

Idea of the Proof:

- Show that q and r both **exist**.
  - Create a generalized set of remainders.
  - Utilize the Well Ordering Principle to prove q and r exist in this set.
- Show that q and r are both **unique** integer values.
  - Introduce two new integers, r' and q'.

Idea of the Proof of the Division Algorithm

Idea of the Proof:

- Show that q and r both **exist**.
  - Create a generalized set of remainders.
  - Utilize the Well Ordering Principle to prove q and r exist in this set.
- Show that q and r are both **unique** integer values.
  - Introduce two new integers, r' and q'.
  - Show $r' = r$ and $q' = q$.

Proof of the Division Algorithm

- Let $S = \{a - bx | x \in \mathbb{Z}, a - bx \geq 0\}$.

Proof of the Division Algorithm

- Let $S = \{a - bx | x \in \mathbb{Z}, a - bx \geq 0\}$.
- Consider our previous example, $26 \div 7$ where $a = 26$ and $b = 7$.

Proof of the Division Algorithm

- Let $S = \{a - bx | x \in \mathbb{Z}, a - bx \geq 0\}$.
- Consider our previous example, $26 \div 7$ where $a = 26$ and $b = 7$.

x	$a - bx$
x	$26 - 7x$
-2	40
-1	33
0	26
1	19
2	12
3	5

Proof of the Division Algorithm

- Let $S = \{a - bx | x \in \mathbb{Z}, a - bx \geq 0\}$.
- Consider our previous example, $26 \div 7$ where $a = 26$ and $b = 7$.

x	$a - bx$
x	$26 - 7x$
-2	40
-1	33
0	26
1	19
2	12
3	5

- Notice that $r = 5$ is the smallest element in S and $x = q = 3$.

Proof of the Division Algorithm

- Let $S = \{a - bx | x \in \mathbb{Z}, a - bx \geq 0\}$.

Proof of the Division Algorithm

- Let $S = \{a - bx | x \in \mathbb{Z}, a - bx \geq 0\}$.
- Claim: $S \neq \emptyset$.

Proof of the Division Algorithm

- Let $S = \{a - bx | x \in \mathbb{Z}, a - bx \geq 0\}$.
- Claim: $S \neq \emptyset$.
  - Case 1: Assume $a \geq 0$.

Proof of the Division Algorithm

- Let $S = \{a - bx | x \in \mathbb{Z}, a - bx \geq 0\}$.
- Claim: $S \neq \emptyset$.
  - Case 1: Assume $a \geq 0$.
    Let $x = 0$. Then $a - bx = a - b(0) = a \in S$.

Proof of the Division Algorithm

- Let $S = \{a - bx | x \in \mathbb{Z}, a - bx \geq 0\}$.
- Claim: $S \neq \emptyset$.
  - Case 1: Assume $a \geq 0$.
    Let $x = 0$. Then $a - bx = a - b(0) = a \in S$.
  - Case 2: Assume $a < 0$.

Proof of the Division Algorithm

- Let $S = \{a - bx | x \in \mathbb{Z}, a - bx \geq 0\}$.
- Claim: $S \neq \emptyset$.
  - Case 1: Assume $a \geq 0$.
    Let $x = 0$. Then $a - bx = a - b(0) = a \in S$.
  - Case 2: Assume $a < 0$.
    Let $x = a$. Then
    $a - bx = a - b(a) = a(1 - b) \geq 0 \Rightarrow a(1 - b) \in S$.

Proof of the Division Algorithm

- Let $S = \{a - bx | x \in \mathbb{Z}, a - bx \geq 0\}$.
- Claim: $S \neq \emptyset$.
  - Case 1: Assume $a \geq 0$.
    Let $x = 0$. Then $a - bx = a - b(0) = a \in S$.
  - Case 2: Assume $a < 0$.
    Let $x = a$. Then
    $a - bx = a - b(a) = a(1 - b) \geq 0 \Rightarrow a(1 - b) \in S$.
- Both cases result in S being nonempty.

Proof of the Division Algorithm

- By the Well Ordering Principle, S has a smallest element.

Proof of the Division Algorithm

- By the Well Ordering Principle, S has a smallest element.
- Let $r = min(S) \Rightarrow a = bq + r \Rightarrow r = a - bq$.

Proof of the Division Algorithm

- By the Well Ordering Principle, S has a smallest element.
- Let $r = min(S) \Rightarrow a = bq + r \Rightarrow r = a - bq$.
- Suppose for the sake of contradiction that $r \geq b$.

Proof of the Division Algorithm

- By the Well Ordering Principle, S has a smallest element.
- Let $r = min(S) \Rightarrow a = bq + r \Rightarrow r = a - bq$.
- Suppose for the sake of contradiction that $r \geq b$.
- Then

$$\begin{aligned} r &= a - bq & &\geq b \\ r - b &= a - bq - b & & \\ r - b &= a - b(q-1) & & \\ r - b &= a - b(q-1) & &\geq b - b = 0 \end{aligned}$$

Proof of the Division Algorithm

- By the Well Ordering Principle, S has a smallest element.
- Let $r = min(S) \Rightarrow a = bq + r \Rightarrow r = a - bq$.
- Suppose for the sake of contradiction that $r \geq b$.
- Then

$$\begin{aligned} r &= a - bq && \geq b \\ r - b &= a - bq - b && \\ r - b &= a - b(q-1) && \\ r - b &= a - b(q-1) && \geq b - b = 0 \end{aligned}$$

- So, $r - b \in S$ but $r - b < r$. Contradiction!

Proof of the Division Algorithm

- By the Well Ordering Principle, S has a smallest element.
- Let $r = min(S) \Rightarrow a = bq + r \Rightarrow r = a - bq$.
- Suppose for the sake of contradiction that $r \geq b$.
- Then

$$
\begin{aligned}
r &= a - bq && \geq b \\
r - b &= a - bq - b && \\
r - b &= a - b(q-1) && \\
r - b &= a - b(q-1) && \geq b - b = 0
\end{aligned}
$$

- So, $r - b \in S$ but $r - b < r$. Contradiction!
- Thus, r and q both exist.

Proof of the Division Algorithm

- Suppose there exist integers r, r', q, q' such that $a = bq + r$, $a = bq' + r'$, $0 \leq r < b$, and $0 \leq r' < b$.

Proof of the Division Algorithm

- Suppose there exist integers r, r', q, q' such that $a = bq + r$, $a = bq' + r'$, $0 \leq r < b$, and $0 \leq r' < b$.
- Then

$$\begin{aligned} bq + r &= bq' + r' & \text{Assume } r' \geq r. \\ bq - bq' &= r' - r \\ b(q - q') &= r' - r \end{aligned}$$

Proof of the Division Algorithm

- Suppose there exist integers r, r', q, q' such that $a = bq + r$, $a = bq' + r'$, $0 \leq r < b$, and $0 \leq r' < b$.
- Then

$$\begin{aligned} bq + r &= bq' + r' \qquad \text{Assume } r' \geq r. \\ bq - bq' &= r' - r \\ b(q - q') &= r' - r \end{aligned}$$

- Notice the left hand side is a multiple of b and $0 \leq r' - r < b$ on the right hand side.

Proof of the Division Algorithm

- Suppose there exist integers r, r', q, q' such that $a = bq + r$, $a = bq' + r'$, $0 \leq r < b$, and $0 \leq r' < b$.
- Then

$$\begin{aligned} bq + r &= bq' + r' \qquad\qquad \text{Assume } r' \geq r. \\ bq - bq' &= r' - r \\ b(q - q') &= r' - r \end{aligned}$$

- Notice the left hand side is a multiple of b and $0 \leq r' - r < b$ on the right hand side.
- Thus, $b(q - q') = r' - r = 0$ so $r = r'$ and since $b > 0, q = q'$. $\square$

Application of the Division Algorithm

- The Division Algorithm is a key tool in the study of number theory and is the basis of various other theorems and methods.

Application of the Division Algorithm

- The Division Algorithm is a key tool in the study of number theory and is the basis of various other theorems and methods.
- **The Euclidean Algorithm** is an algorithm that allows one to find the greatest common divisor of two integers by repeatedly performing the Division Algorithm.

Application of the Division Algorithm

- The Division Algorithm is a key tool in the study of number theory and is the basis of various other theorems and methods.
- **The Euclidean Algorithm** is an algorithm that allows one to find the greatest common divisor of two integers by repeatedly performing the Division Algorithm.
  - The GCD of two integers a and b is the greatest positive integer d that is a divisor of both a and b, denoted $d = \gcd(a, b)$.

Application of the Division Algorithm

- The Euclidean Algorithm spells out the following steps to compute $d = \gcd(a, b)$:

Application of the Division Algorithm

- The Euclidean Algorithm spells out the following steps to compute $d = \gcd(a, b)$:

$$
\begin{aligned}
a &= bq_1 + r_1 \\
b &= r_2 q_2 + r_2 \\
r_1 &= r_2 q_3 + r_3 \\
&\vdots \\
r_{n-2} &= r_{n-1} q_n + r_n \\
r_{n-1} &= r_n q_{n+1}
\end{aligned}
$$

where $r_n = d$.

Application of the Division Algorithm

- The Extended Euclidean Algorithm is an expansion of the Euclidean Algorithm that results in $d = ra + sb$.

Application of the Division Algorithm

- The Extended Euclidean Algorithm is an expansion of the Euclidean Algorithm that results in $d = ra + sb$.
- The Euclidean Algorithm, combined with the Extended's integers r, s, both aid in creating and deciphering keys in RSA cryptography.

Application of the Division Algorithm

- The Extended Euclidean Algorithm is an expansion of the Euclidean Algorithm that results in $d = ra + sb$.
- The Euclidean Algorithm, combined with the Extended's integers r, s, both aid in creating and deciphering keys in RSA cryptography.
    - RSA cryptography is used to enhance the security of data encryption for emails, transactions, etc.

Conclusion

- Although simple, the Division Algorithm serves several purposes in the realm of both theoretical and applied mathematics.

Conclusion

- Although simple, the Division Algorithm serves several purposes in the realm of both theoretical and applied mathematics.
  - It's the basis of several number theory concepts, including the Euclidean Algorithm, which was created around 300 BC.

Conclusion

- Although simple, the Division Algorithm serves several purposes in the realm of both theoretical and applied mathematics.
  - It's the basis of several number theory concepts, including the Euclidean Algorithm, which was created around 300 BC.
  - Its uses can also extend to significant real-world problems, like online security.

References

JUDSON, T.
Abstract Algebra: Theory and Applications.
Virginia Commonwealth University Mathematics, 2009.

LEHMAN, E., LEIGHTON, T., AND MEYER, A. R.
Mathematics for computer science.
Tech. rep., Technical report, 2006. Lecture notes, 2010.

OBERMANN, S., AND FLYNN, M. J.
An analysis of division algorithms and implementations.
Tech. rep., Technical Report CSL-TR-95-675, Stanford University, 1995.

SHALLIT, J.
Origins of the analysis of the euclidean algorithm.
Historia Mathematica 21, 4 (1994), 401–419.