UECM1304 Discrete Mathematics with Applications Topic 3: Elementary Number Theory and Methods of Proof

Lecturer: Dr. Liew How Hui
Email: liewhh@utar.edu.my

16 hours

This topic explores the *theory of numbers*, an important branch of mathematics concerning the properties of integers. *Integers* are *central* to discrete mathematics because integer functions are used as the measurement of time-complexity for computer algorithms and the properties of integers are crucial in the encryption system of secure communications. In addition, the random number generator used in stochastic simulation also has its foundation build on top of the theory of numbers.

Contents

CLO3: Demonstrate various proof-techniques.

§3.1 Formal Characterisation of Numbers

This section discusses how to formally express the following number sets that we often encounter in mathematics.

Symbol	Set
$\mathbb{N}$	Set of all natural numbers, i.e. $\{0, 1, 2, \cdots\}$
$\mathbb{Z}$	Set of all integers, i.e. $\{\cdots, -2, -1, 0, 1, 2, \cdots\}$
$\mathbb{Q}$	Set of all rational numbers, i.e. $\{\frac{p}{q} : p, q \in \mathbb{Z}, q \neq 0\}$
$\mathbb{R}$	Set of all real numbers, i.e. $\{\pm i_n \cdots i_2 i_1.d_1 d_2 \cdots : i_j, d_i \in \mathbb{N}\}$

They are related by $\mathbb{N} \subset \mathbb{Z} \subset \mathbb{Q} \subset \mathbb{R}$ “semantically”. To differentiate positive, negative and nonzero numbers, we introduce the following notations:

- $\mathbb{R}^+ := \{x \in \mathbb{R} : x > 0\}$, $\mathbb{R}^- := \{x \in \mathbb{R} : x < 0\}$, $\mathbb{R}^* := \{x \in \mathbb{R} : x \neq 0\}$.
- $\mathbb{Q}^+ := \{x \in \mathbb{Q} : x > 0\}$, $\mathbb{Q}^- := \{x \in \mathbb{Q} : x < 0\}$, $\mathbb{Q}^* := \{x \in \mathbb{Q} : x \neq 0\}$.
- $\mathbb{Z}^+ := \{1, 2, 3, \cdots\}$, $\mathbb{Z}^- := \{-1, -2, -3, \cdots\}$, $\mathbb{Z}^* := \{\pm 1, \pm 2, \cdots\}$.

A set of formal propositions, called ***axioms***, are used to characterise the set of numbers mentioned above.

§3.1.1 Natural Numbers

Peano axioms define the arithmetical properties of natural numbers, usually represented as $\mathbb{N}$.

The symbols for the axioms includes a constant symbol 0 and a unary function symbol S and an equal sign “=”.

1. $0 \in \mathbb{N}$
2. $\forall x \exists y (S(x) = y)$ . . . . every natural number has a “next” natural number.
3. $\exists 0 \forall x (\sim (S(x) = 0))$ . . . . there is a 0 which has no “previous” natural number.
4. $\forall x (x = x)$ . . . . reflexive
5. $\forall x \forall y (x = y \to y = x)$ . . . . symmetric
6. $\forall x \forall y \forall z (x = y \wedge y = z \to x = z)$ . . . . transitive
7. $\forall x \forall y (S(m) = S(n) \to m = n)$
8. $\forall y (\phi(0, y) \wedge \forall x (\phi(x, y) \Rightarrow \phi(S(x), y)) \Rightarrow \forall x \phi(x, y))$ . . . . mathematical induction

$\mathbb{N}$ is implemented as `Nat` type in Lean 4's `Init/Prelude.lean`

```
inductive Nat where
  | zero : Nat                -- same as 0
  | succ (n : Nat) : Nat     -- same as n+1
```

With the above definitions, it is possible to obtain the "ordered semiring" properties of $\mathbb{N}$ below.

A1. $\forall x \forall y(x + y = y + x)$ .... `Nat.add_comm`

A2. $\forall x \forall y \forall z((x + y) + z = x + (y + z))$ .... associative law

A3. $\forall x(0 + x = x)$; .... identity

M1. $\forall x \forall y(x \cdot y = y \cdot x)$ .... commutative law

M2. $\forall x \forall y \forall z((x \cdot y) \cdot z = x \cdot (y \cdot z))$ .... associative law

M3. $\forall x(1 \cdot x = x) \wedge (0 \cdot x = 0)$ .... identity

D. $\forall x \forall y \forall z(x \cdot (y + z) = x \cdot y + x \cdot z)$. .... distributive law

L1. $\forall x \sim (x < x)$

L2. $\forall x \forall y((x < y) \vee (x = y) \vee (y < x))$ .... total ordering

L3. $\forall x \forall y \forall z(((x < y) \wedge (y < z)) \to x < z)$ .... transitivity

L4. $\forall x \forall y \forall z((x < y) \to (z + x < z + y))$

L5. $\forall x \forall y \forall z(((x < y) \wedge (0 < z)) \to z \cdot x < z \cdot y)$.

L6. $\forall x \forall y((x < y) \to \exists z(x + z = y))$

L7. $(0 < 1) \wedge (\forall x((x > 0) \to (x = 1 \vee x > 1))$.

L8. $\forall x(x = 0 \vee 0 < x)$.

I. $\phi(0) \wedge (\forall k(\phi(k) \to \phi(k + 1))) \to \forall n \phi(n)$ .... Induction Principle

Here ϕ is any well-form formula.

A1–D in Lean 4's `Init/Data/Nat/Basic.lean`

```
variable (x y z : Nat)
example : x + y = y + x := Nat.add_comm x y
example : (x + y) + z = x + (y + z) := Nat.add_assoc x y z
example : 0 + x = x := Nat.zero_add x
example : x + 0 = x := Nat.add_zero x
example : x * y = y * x := Nat.mul_comm x y
example : (x * y) * z = x * (y * z) := Nat.mul_assoc x y z
example : x * 1 = x := Nat.mul_one x
example : 1 * x = x := Nat.one_mul x
example : 0 * x = 0 := Nat.zero_mul x
example : x * (y + z) = x * y + x * z := Nat.mul_add x y z
example : x * (y + z) = x * y + x * z := Nat.left_distrib x y z
example : (x + y) * z = x * z + y * z := Nat.add_mul x y z
example : (x + y) * z = x * z + y * z := Nat.right_distrib x y z
theorem L3 : (x < y) /\ (y < z) -> x < z :=
  fun h => Nat.lt_trans h.left h.right
theorem L4 : (x < y) -> (z + x < z + y) :=
  fun h => Nat.add_lt_add_left  h _
#check 0 + 1 = 1    -- Print the type of the expression
#eval 0 + 1 = 1     -- Perform calculation to decide the value
```

From the Induction Principle, mathematicians have obtained the well-ordering principle stated below.

Theorem 3.1.1 (Well-Ordering Property of $\mathbb{N}$)**.** *If $S \subset \mathbb{N}$ and $S \neq \emptyset$, then there is an element $m \in S$ such that $m \leq k$ for all $k \in S$.*

§3.1.2 Integers: Even, Odd, Prime, Composite

A set of integers together with the mathematical operations makes it into a *commutative ordered ring with unit* whose positive elements are well-ordered.

Definition 3.1.2. The *commutative ordered ring with unit* F is a set equipped with constants 0, 1, addition $+$, multiplication $\cdot$ and less than relation $<$ such that

A1. $\forall x \forall y (x + y = y + x)$; .... commutativity

A2. $\forall x \forall y \forall z ((x + y) + z = x + (y + z))$; .... associativity

A3. $\forall x (0 + x = x)$; .... identity

A4. $\forall x \exists y (x + y = 0)$; .... additive inverse

M1. $\forall x \forall y (x \cdot y = y \cdot x)$; .... commutativity

M2. $\forall x \forall y \forall z ((x \cdot y) \cdot z = x \cdot (y \cdot z))$; .... associativity

M3. $(1 \neq 0) \wedge \forall x (1 \cdot x = x)$; .... identity

D. $\forall x \forall y \forall z (x \cdot (y + z) = x \cdot y + x \cdot z)$; .... distributivity

O1. $\forall x \forall y ((x = y) \wedge \sim (y < x) \wedge \sim (x < y) \vee \sim (x = y) \wedge (y < x) \wedge \sim (x < y) \vee \sim (x = y) \wedge \sim (y < x) \wedge (x < y))$; .... total order

O2. $\forall x \forall y \forall z ((x < y) \wedge (y < z) \to (x < z))$; .... transitivity

O3. $\forall x \forall y \forall z (x < y \to (z + x < z + y))$;

O4. $\forall x \forall y \forall z (((x < y) \wedge (0 < z)) \to z \cdot x < z \cdot y)$

Definition 3.1.3. The *integer set* $\mathbb{Z}$ is a commutative ordered ring with unit such that

I. $\phi(0) \wedge (\forall k (\phi(k) \to \phi(k+1))) \to \forall n \phi(n)$ .... Induction Principle

$\mathbb{Z}$ is implemented as `Int` type in Lean 4's `Init/Data/Int/Basic.lean`

```
inductive Int : Type where
  | ofNat   : Nat -> Int  -- n = 0, 1, 2, ...
  | negSucc : Nat -> Int  -- negatives are -[1+n]
def negOfNat : Nat -> Int
  | 0      => 0
  | succ m => negSucc m
```

A1–D in Lean 4's `Init/Data/Int/`

```
variable (x y z : Int)
theorem A1 : x + y = y + x := Int.add_comm x y
theorem A2 : (x + y) + z = x + (y + z) := Int.add_assoc x y z
theorem A3 : 0 + x = x := Int.zero_add x
theorem A3r: x + 0 = x := Int.add_zero x
theorem M1 : x * y = y * x := Int.mul_comm x y
theorem M2 : (x * y) * z = x * (y * z) := Int.mul_assoc x y z
theorem M3 : 1 * x = x := Int.one_mul x
theorem M3r: x * 1 = x := Int.mul_one x
theorem D  : x * (y + z) = x * y + x * z := Int.mul_add x y z
theorem Dr : (x + y) * z = x * z + y * z := Int.add_mul x y z
theorem O2 : (x < y) /\ (y < z) -> x < z := fun h => Int.lt_trans h.left h.right
theorem O3 : (x < y) -> (z + x < z + y) := fun h => Int.add_lt_add_left h _
```

Definition 3.1.4. Let n be an integer. n is *even* if there is an integer k such that $n = 2k$. n is *odd* if an integer k such that $n = 2k + 1$.

Definition of Even and Odd in Lean 4

```
def is_even (n : Int) := exists k, n = 2*k
def is_odd  (n : Int) := exists k, n = 2*k+1
```

Theorem 3.1.5. *Every integer is either odd or even but not both.*

Example 3.1.6. Use the definitions of even and odd to justify your answers to the following questions.

1. Is 0 even?

2. Is -501 odd?

3. If a and b are integers, is $8ab^3 + 6$ even?

Definition 3.1.7. The *absolute value* of an integer n, $|n| := \begin{cases} n, & 0 \leq n, \\ -n, & n < 0. \end{cases}$

Definition 3.1.8. A integer n is called a *prime (number)* if $n > 1$ and "$\forall r \forall s((r \in \mathbb{N} \wedge s \in \mathbb{N} \wedge n = r \cdot s) \rightarrow ((r = 1) \vee (s = 1)))$". n is called *composite* if $n > 1$ and there are positive integers r and s such that $n = r \cdot s$ and $r \neq 1$ and $s \neq 1$, i.e. those numbers that have more than two factors. A negative integer n is *composite* if $|n|$ is composite.

Implementation of Even, Odd, Prime Checking Functions in Standard ML

```
fun isEven(n : int): bool = (n mod 2 = 0)
fun isOdd (n : int): bool = (n mod 2 <> 0)
(*
0 is neither prime nor composite. Since any number times zero
equals zero, there are an infinite number of factors for
a product of zero.
1 has only 1 factor. For a number to be classified as
a prime number, it should have exactly two factors.
*)
fun isPrime(n: int): bool =
  let fun noDivisorsAbove(m: int) =
    if n mod m = 0 then false
    else if m*m >= n then true
     else noDivisorsAbove(m+1)
  in
    n > 1 andalso (n=2 orelse noDivisorsAbove(2))
  end
fun isComposite(n: int): bool =
  let fun hasDivisorsAbove(m: int) =
    if n mod m = 0 then true
    else if m*m >= n then false
         else hasDivisorsAbove(m+1)
  in
    (abs n) > 2 andalso hasDivisorsAbove(2)
  end
```

Theorem 3.1.9. *Every integer greater than one is either a prime or a composite but not both.*

Example 3.1.10. Let $E(n)$ be "n is even", $P(n)$ be "n is prime" and $C(n)$ be "n is composite."

1. Translate the following into English without using the symbols $\exists$ and $\forall$.

   (a) $\exists n(P(n) \land E(n))$ .... ______________________________

   (b) $\forall n(E(n) \lor \sim P(n))$ .... ______________________________

   (c) $\sim \forall n(E(n) \lor P(n))$ .... ______________________________

2. Determine the truth value of these statements.

   (a) $P(13) \to \sim E(13)$ .... □

   (b) $P(2) \to \sim E(2)$ .... □

   (c) $P(0)$ .... □

   (d) $C(0)$ .... □

   (e) $P(1)$ .... □

   (f) $C(1)$ .... □

   (g) $C(-3)$ .... □

   (h) $C(-4)$ .... □

Example 3.1.11. 1. Write the first 6 prime numbers.

2. Write the first 6 non-negative composite numbers.

Example 3.1.12. Explain how the function `isEven 3` work.

Solution: 3 (mod 2) = 1 = 0 is false.

Example 3.1.13. Explain how the function `isOdd 3` work.

Example 3.1.14. Explain how the function `isPrime 17` work.

Solution:

isPrime 17 = 17 > 1 and noDivisorAbove(2)
noDivisorAbove(2) = 17 mod 2 $\neq$ 0 and 2*2 < 17 calls noDivisorAbove(3)
noDivisorAbove(3) = 17 mod 3 $\neq$ 0 and 3*3 < 17 calls noDivisorAbove(4)
noDivisorAbove(4) = 17 mod 4 $\neq$ 0 and 4*4 < 17 calls noDivisorAbove(5)
noDivisorAbove(5) = 17 mod 5 $\neq$ 0 and 5*5 $\geq$ 17 returns true
So isPrime 17 = true and true = true

Example 3.1.15. Explain how the function `isComposite 16` work.

Definition 3.1.16. A *counterexample* to the universal statement $\forall x P(x)$ is a value c for which $v(P(c)) = F$.

Example 3.1.17. Determine the truth value of the following universal statements. If the universal statement is false, suggest a counterexample to the universal statement.

1. For every $x \in \{-2, 0, 4, 6, 8\}$, x^2 is even.

Note: The technique used to show the truth of the universal statement in this example is called the *method of exhaustion*. This technique is used when the domain has finite elements.

2. $\forall x(x \in \mathbb{Z} \to x + 1 < 4)$.

3. $\forall x(x \in \mathbb{Z}^+ \to (x - 1 \text{ is nonnegative}))$.

Example 3.1.18. Determine the truth value of each of the following existential statement.

1. There is an integer x such that a prime number x is an even number. .... ________________
2. $\exists x(x \in \mathbb{Z} \wedge x = x + 1)$. .... ________________

Example 3.1.19 (Tutorial 3, Q1)**.** Assuming that m and n are particular integers, use the definitions of even, odd, prime and composite to answer the following questions.

1. If $m > n > 0$, is $m^2 - n^2$ composite?
2. Is $6m + 10n$ even?
3. Is $10mn + 13$ odd?
4. If $m > 0$ and $n > 0$, is $m^2 + 2mn + n^2$ composite?

§3.1.3 Rational Numbers and Real Numbers

The set of rational numbers $\mathbb{Q}$ is the "smallest" Archimedean ordered field and the set of real numbers $\mathbb{R}$ is an Archimedean ordered field which is Cauchy complete. Hence, we will first define the notion of ordered field.

Definition 3.1.20. A set F (the domain for the predicates) is called an ***ordered field*** if it is a commutative ordered ring with unit (Definition 3.1.2) and satisfies

M5. $\forall x(x \neq 0 \to (\exists y, y \cdot x = 1))$; .... multiplicative inverse

Definition 3.1.21. We say an ordered field is ***Archimedean*** if for each $\epsilon \in F$ such that $\epsilon > 0$, there is positive integer N such that $1 < N\epsilon$.

Definition 3.1.22. The rational number set $\mathbb{Q}$ is the smallest Archimedean ordered field. The member of $\mathbb{Q}$ is called ***rational number***.

Theorem 3.1.23. *A rational number $r \in \mathbb{Q}$ can be expressed as a quotient of two integers with a nonzero denominator, i.e. there are integers a and $b \neq 0$ such that $\frac{a}{b}$.*

Theorem 3.1.24. *Every integer is a rational number.*

Proof. For any integer n, $n = n/1$. □

Example 3.1.25. Which of the following are rational numbers?

- $-(4/35)$
- 0.211
- $0.\dot{1}\dot{2}$.

Example 3.1.26. If m and n are integers and neither m nor n is zero, is $\dfrac{m+n}{mn}$ a rational number?

Definition 3.1.27. Suppose F is an ordered field and $E \subset F$. If there exists a $\beta \in F$ such that $x \leq \beta$, $\forall x \in E$, we say that E is *bounded above* and call β an *upper bound* of E. If the upper bound β of E also satisfies the following property

- If $\gamma < \beta$ then γ is not an upper bound of E.

Then β is called the *least upper bound* of E or the *supremum of* E and we write $\beta = \sup E$.

Axiom 3.1.28 (Completeness Axiom). .... `completeness`

Let F be an ordered field and $E \subset F$. If $E \neq \emptyset$, and E is bounded above, then $\sup E$ exists in F.

Definition 3.1.29. The real number set $\mathbb{R}$ is an ordered field that satistifies the Completeness Axiom. The members of $\mathbb{R}$ are called *real numbers*.

Theorem 3.1.30. *A rational number is a real number.*

Definition 3.1.31. An *irrational* number is a real number that is not rational.

By definition, every real number is either rational or irrational but not both.

Example 3.1.32. Determine the truth value of the following universal statements. If the universal statement is false, suggest a counterexample to the universal statement.

1. $\forall x((x \in \mathbb{R} \land 0 \leq x \land x \leq 1) \to x < 5x)$

2. $\forall x(x \in \mathbb{R} \to x^2 + 1 = 0)$

3. $\forall x(x \in \mathbb{R} \to x^2 - 1 = (x+1)(x-1))$

Example 3.1.33. Determine the truth value of each of the following existential statement.

1. $\exists x(x \in \mathbb{R} \wedge x \geq 0 \wedge x^2 + 5x + 6 = 0)$ .... ____________________
2. $\exists x(x \in \mathbb{R} \wedge x^2 + 1 = 0)$ .... ____________________
3. $\exists x(x \in \mathbb{R} \wedge x^3 = x^2 - 2)$ .... ____________________
4. $\exists x(x \in \mathbb{R} \wedge (x^2 - 2 = (x + \sqrt{2})(x - \sqrt{2})))$ .... ____________________
5. $\exists x(x \in \mathbb{R} \wedge (\frac{x}{x^2+1} = \frac{2}{5}))$ .... ____________________

Example 3.1.34 (Tutorial 3, Q2)**.** (a) Assume that $a \neq 0$ and $b \neq 0$ are both integers. Is $(b-a)/(ab^2)$ a rational number?

(b) Assume that a and $b > 0$ are both integers. Is $(5a + 12b)/4b$ a rational number?

Example 3.1.35 (Tutorial 3, Q3)**.** Suppose a, b, c and d are integers and $a \neq c$. Suppose also x is a real number that satisfies the equation

$$\frac{ax+b}{cx+d} = 1. \qquad (**)$$

Is x rational?

Example 3.1.36 (Tutorial 3, Q4)**.** Is the following argument valid?

Any sum of two rational numbers is rational.
The sum $r + s$ is rational.

Therefore the numbers r and s are both rational.

§3.2 Methods of Proof

To **confirm** a mathematical statement is **true**, we need to **prove** it.

Let $A_1, \cdots, A_n$ be the **axioms** of **mathematical objects** (e.g. $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$).

We can derive theorems $T_1, \cdots, T_m$ from the axioms $A_1, \cdots, A_n$. For a new mathematical statement ψ, if we can prove it using the axioms and theorems, then it can be expressed as an argument below:

$$A_1,\ \cdots,\ A_n,\ T_1,\ \cdots,\ T_m \vdash \psi. \tag{3.1}$$

The techniques of *direct proof* and *proof by contradiction* are usually used to show the argument (3.1) syntactically.

For a new mathematical statement in conditional form, the argument can be expressed as

$$A_1,\ \cdots,\ A_n,\ T_1,\ \cdots,\ T_m \vdash (\phi \to \psi) \tag{3.2}$$

where ϕ and ψ be two formulas. The techniques of *direct proof*, *contrapositive proof* and *proof by contradiction* are usually used to show the argument (3.1) syntactically. To know more about the different aspects of mathematical proofs, one may want to read Ording [2019].

§3.2.1 Direct Proof

The ***direct proof*** of (3.2) **starts from** axioms, theorems and ϕ **to reach** ψ.

Example 3.2.1. 12 is an even number.

Proof: $12 = 2 \times 6$ and by definition, 12 is even.

Proving in Lean 4

```
example : is_even (12) :=
  have h : Int.ofNat 12 = 2*6 := rfl  -- calculate & check equality
  Exists.intro 6 h                    -- k = 6
```

Example 3.2.2. Prove that "there is an even integer n that can be written in 2 ways as a sum of 2 prime numbers."

Proof: Let $n = 10$. Then $10 = 5 + 5 = 3 + 7$ and 3, 5 and 7 are all prime numbers.

Let $P(n)$ be the predicate "n is prime". Then take $n = 10$, $p_1 = p_2 = 5$, $p_3 = 3$, $p_4 = 7$ and

$$\begin{array}{rl}
 & \underline{10 = 2 \cdot 5} \land \underline{5 \neq 3 \land 5 \neq 7 \land 5 \neq 7} \\
 & \underline{\land P(5) \land P(5) \land P(3) \land P(7)} \land \underline{10 = 5 + 5 \land 10 = 3 + 7}) \\
\hline
\therefore & \exists n,\ (\underline{\exists k(n = 2k)} \land \underline{\exists p_1 \exists p_2 \exists p_3 \exists p_4 (p_1 \neq p_3 \land p_1 \neq p_4 \land p_2 \neq p_4} \\
 & \underline{\land P(p_1) \land P(p_2) \land P(p_3) \land P(p_4)} \land \underline{n = p_1 + p_2 \land n = p_3 + p_4})
\end{array}$$

Due to the difficulty in manipulating formal proofs using appropriate tactics, we usually do not use formal proofs in deriving results in mathematics.

Example 3.2.3. Use the **method of exhaustion** (useful for domain with finite values) to prove the following statements:

1. If n is even and $4 \leq n \leq 20$, then n can be written as a sum of 2 prime numbers.

Proof:

$4=2+2$ $8=3+5$ $12=5+7$ $16=5+11$ $20=7+13$

$6=3+3$ $10=5+5$ $14=11+3$ $18=7+11$

2. Every even positive integer n which are less than 26 can be written as a sum of less than or equal to 3 perfect squares.

Example 3.2.4 (Tutorial 3, Q7)**.** Prove the following universal statement by using method of exhaustion.

For each integer n with $1 \leq n \leq 10$, $n^2 - n + 11$ is a prime number.

Example 3.2.5. Suppose r and s are integers. Prove that "there is an integer k such that $22r + 18s = 2k$".

Proof: Let $k = 11r + 9s$. Then k is an integer because it is a sum of products of integers and $2k = 2(11r + 9s) = 22r + 18s$.

Example 3.2.6. Prove that the sum of any two even integers is even.

Remark: In this case you might imagine some pairs of even integers, say 2 + 4, 6 + 10, 12 + 12, 28 + 54, and mentally check that their sums are even. However, since you cannot possibly check all pairs of even numbers, you cannot know for sure that the statement is true in general by checking its truth in these particular instances. Many properties hold for a large number of examples and yet fail to be true in general.

To prove this statement in general, you need to show that no matter what even integers are given, their sum is even.

Proof: Suppose m and n are any even integers.
By definition of even, $m = 2r$ and $n = 2s$ for some integers r and s.
Then $m + n = 2r + 2s = 2(r + s)$.
Let $k = r + s$. k is an integer because it is a sum of integers r and s.
Hence $m + n = 2k$ where k is an integer. It follows by definition of even that $m + n$ is even.

Proving in Lean 4

```
theorem even_plus_even (h1 : is_even a) (h2 : is_even b) : is_even (a
+ b) :=
  Exists.elim h1 (fun w1 (hw1 : a = 2 * w1) =>
  Exists.elim h2 (fun w2 (hw2 : b = 2 * w2) =>
    Exists.intro (w1 + w2)
      (calc a + b
        _ = 2 * w1 + 2 * w2 := by rw [hw1, hw2]
        _ = 2 * (w1 + w2)   := by rw [Int.mul_add])))
```

Example 3.2.7. Prove that the product of any two even integers is even.

Proof: Suppose m and n are any even integers.

By definition of even, $m = 2r$ and $n = 2s$ for some integers r and s.

Then $mn = (2r) \cdot (2s) = 2(2rs)$.

Let $k = 2rs$. Note that k is an integer because it is a product of integers.

Hence $mn = 2k$ where k is an integer.

It follows by definition of even that mn is even.

Proving in Lean 4

```
theorem even_mul_even (h1 : is_even a) (h2 : is_even b) : is_even (a
* b) :=
  Exists.elim h1 (fun w1 (hw1 : a = 2 * w1) =>
    Exists.intro (w1*b)
      (calc a*b
        _ = (2 * w1) * b := by rw [hw1]
        _ = 2 * (w1 * b) := by rw [Int.mul_assoc]))
```

Example 3.2.8. Prove that the sum of any two odd integers is even.

Proof.

Example 3.2.9 (Tutorial 3, Q8)**.** Prove the following universal statements:

1. For all integers n, if n is odd then n^2 is odd.
2. If n is any odd integer, then $(-1)^n = -1$.

Example 3.2.10 (Tutorial 3, Q6)**.** Prove the following existential statements:

1. There are distinct integers m and n such that $1/m + 1/n$ is an integer.
2. There are real numbers a and b such that $\sqrt{a+b} \neq \sqrt{a} + \sqrt{b}$.
3. There is an integer n such that $2n^2 - 5n + 2$ is prime.

Example 3.2.11 (Tutorial 3, Q5)**.** Use the rules of inference and real number axioms to prove that $/ \therefore \forall x(3 < x \rightarrow 25 < x^2 + 5x + 2)$.

Theorem 3.2.12. *Prove that the sum of any two rational numbers is rational.*

Proof.

A ***corollary*** is a statement whose truth can be immediately deduced from a theorem that has already been proved.

Corollary 3.2.13. *The double of a rational number is rational.*

§3.2.2 Proof by Contraposition/Contrapositive Proof

Proof by contraposition or ***contrapositive proof*** for a conditional statement $\phi \to \psi$ is based on the logical equivalence $\phi \to \psi \equiv \sim \psi \to \sim \phi$. Formally, contrapositive proof of (3.2) can be expressed as

$$A_1, \ \cdots, \ A_n, \ T_1, \ \cdots, \ T_m \ \vdash \sim \psi \to \sim \phi.$$

The outline of the proof is given below:

1. Rewrite the statement $\phi \to \psi$ to be proved in the contrapositive form $\sim \psi \to \sim \phi$
2. Prove the contrapositive form using direct proof.

Example 3.2.14. Prove that for all integers n, if n^2 is even then n is even.

Proof: In contrapositive form: For all integers n, if n is odd then n^2 is odd.

Suppose n is any odd integer. Then $n = 2k+1$ for some integer k. $n^2 = (2k+1)^2 = 4k^2+4k+1 = 2(2k^2+2k)+1$ So n^2 is odd.

Remark 3.2.15. The proof is similar to the direct proof given in Example 3.2.9.

Example 3.2.16. Prove that for all integers m and n, if $m+n$ is odd, then m is odd or n is odd.

Proof: Suppose that m is even **and** n is even. By definition, $m = 2k_1$ for some k_1 and $n = 2k_2$ for some k_2.

Therefore $m + n = 2k_1 + 2k_2 = 2(k_1 + k_2)$ where $k_1 + k_2$ is an integer. By definition, $m + n$ is even. □

Example 3.2.17. Prove that for all integers n, if $3n + 2$ is odd then n is odd.

Proof.

Example 3.2.18. Prove that for any natural numbers n, a and b, if $n = ab$, then $a \leq \sqrt{n}$ or $b \leq \sqrt{n}$.

Proof.

Example 3.2.19 (Tutorial 3, Q13)**.** Prove the following statements by contraposition.

1. If a product of two positive real numbers is greater than 100, then at least one of the numbers is greater than 10.

2. If a sum of two real numbers is less than 50, then at least one of the numbers is less than 25.

§3.2.3 Proof by Contradiction

Proof by contradiction for (3.1) can be expressed as

$$A_1,\ \cdots,\ A_n,\ T_1,\ \cdots,\ T_m\ \vdash \sim\psi \to \mathrm{F}. \tag{3.3}$$

The *proof by contradiction* for a conditional statement $\phi \to \psi$ can be expressed as

$$A_1,\ \cdots,\ A_n,\ T_1,\ \cdots,\ T_m\ \vdash\ (\phi\wedge\sim\psi) \to \mathrm{F}. \tag{3.4}$$

This technique is very similar to the tableaux method for checking the validity of an argument.

Example 3.2.20. For all integers m and n, if $mn = 1$ then $m = n = 1$ or $m = n = -1$.

This is of the form (3.4).

Domain: $\mathbb{Z}$. m and n are integers and axioms and theorems of integers must the obey.

Hypotheses ϕ: "$mn = 1$".

Conclusion ψ: "$m = n = 1$ or $m = n = -1$".

Proof by contradiction.

Suppose "$mn = 1$" is true but "$(m = 1 \wedge n = 1) \vee (m = -1 \wedge n = -1)$" is false.

So $(m, n \in \mathbb{Z}) \wedge ((m \neq 1) \vee (n \neq 1)) \wedge ((m \neq -1) \vee (n \neq -1))$. This is logically equivalent to

$$(m \neq 1 \wedge m \neq -1) \vee (n \neq 1 \wedge m \neq -1) \vee (m \neq 1 \wedge n \neq -1) \vee (n \neq 1 \wedge n \neq -1).$$

We can classify into the following cases:

1. $m \neq 1 \wedge m \neq -1$:
   (a) $m < -1$:
      i. $n = 0$: $mn = 0$. Contradicting with $mn = 1$.
      ii. $n \leq -1$: $mn > 1$. Contradicting with $mn = 1$.
      iii. $n \geq 1$: $mn < -1$. Contradicting with $mn = 1$.
   (b) $m = 0$: $mn = 0$. Contradicting with $mn = 1$.
   (c) $m > 1$:
      i. $n = 0$: $mn = 0$. Contradicting with $mn = 1$.
      ii. $n \leq -1$: $mn < -1$. Contradicting with $mn = 1$.
      iii. $n \geq 1$: $mn > 1$. Contradicting with $mn = 1$.
2. $n \neq 1 \wedge m \neq -1$:
   (a) $n > 1 \wedge m > -1$: $mn = 0 \vee mn > 1$. Contradicting with $mn = 1$.
   (b) $n > 1 \wedge m < -1$: $mn < -1$. Contradicting with $mn = 1$.
   (c) $n < 1 \wedge m > -1$: $mn = 0 \vee mn < 0$. Contradicting with $mn = 1$.
   (d) $n < 1 \wedge m < -1$: $mn = 0 \vee mn > 1$. Contradicting with $mn = 1$.
3. $m \neq 1 \wedge n \neq -1$: This is similar to case 2., the only difference is m and n are exchanged.
4. $n \neq 1 \wedge n \neq -1$: This is similar to case 1., the only difference is m and n are exchanged.

All situation leads to contradiction. Hence, the conclusion cannot be false and hence the statement is proved.

Example 3.2.21. Show that the rational number $\frac{1}{4}$ is not an integer.

Proof: This is of the form (3.3). We will suppose $\sim \psi$ and then obtain a contradiction (with known results).

Suppose the rational number $\frac{1}{4}$ is an integer. Then

$$4 \times \frac{1}{4} = 1$$

and 1 can be factorised into two integers different from 1 and -1. This is contradicting with Example 3.2.20.

Theorem 3.2.22. *There is no greatest integer.*

Proof: Suppose there is a greatest integer N. Then $n \leq N$ for every integer n.

Let $M = N + 1$. Now M is an integer since it is a sum of integers. Also $N < N + 1 = M$ by O3 from Definition 3.1.2.

Thus M is an integer that is greater than N.

However, N is the greatest integer, so $M < N$. Hence $M < N \wedge N < M$ which is a contradiction violating O1 of Definition 3.1.2.

Thus the supposition is false and "there is no greatest integer" is true.

Example 3.2.23 (Tutorial 3, Q14(a))**.** Use proof by contradiction to prove that there is no greatest even integer.

Theorem 3.2.24. *Using the method of proof by contradiction to prove Theorem 3.1.5, i.e. "there is no integer that is both even and odd."*

Proof: Suppose there is an integer n that is both even and odd.

By definition of even, $n = 2a$ for some integer a. By definition of odd, $n = 2b + 1$ for some integer b.

By equality

$$2a = 2b + 1 \Rightarrow 2(a - b) = 1, \quad a - b \in \mathbb{Z}.$$

This is a contradiction since 1 cannot be factorised as proved in Example 3.2.20.

Example 3.2.25. Use proof by contradiction to prove that for all integers n, if n^2 is even then n is even.

Proof.

Example 3.2.26 (Tutorial 3, Q14(b))**.** Use proof by contradiction to prove that "for all real numbers x and y, if x is irrational and y is rational then $x - y$ is irrational."

§3.2.4 Classical Theorems: Irrationality of $\sqrt{2}$; Infinitude of Primes

There are two classical theorems which are derived using the proof by contradiction (using integer divisibility results from Section 3.5).

Theorem 3.2.27. *$\sqrt{2}$ is irrational.*

Proof: Suppose $\sqrt{2}$ is rational. Then there are integers m and n with no common factors and $n \neq 0$ such that

$$\sqrt{2} = \frac{m}{n}.$$

This implies

$$2 = \frac{m^2}{n^2} \Rightarrow 2n^2 = m^2$$

So m^2 is even and m must be even (otherwise it must be odd, but the square of odd number must be odd) and there is an integer k such that $m = 2k$. Hence

$$2n^2 = m^2 = 4k^2 \Rightarrow n^2 = 2k^2.$$

Now, n^2 is even and so n is even. This implies that both m and n have a common factor of 2, which contradicts the supposition that m and n have no common factors.

Using Theorem 3.2.27, we can show that something exists without finding the actual thing that matches the predicate. This is called the *non-constructive proof* and is demonstrated below.

Example 3.2.28. Prove that "There exist irrational numbers x and y s.t. x^y is rational".

Proof.

Theorem 3.2.29 (Generalisation of Theorem 3.2.27)**.** *For a positive integer k, if $\sqrt{k}$ is not integer, then $\sqrt{k}$ is irrational.*

Remark: In mathematics, a proof by infinite descent is a particular kind of proof by contradiction which relies on the facts that the natural numbers are well ordered and that there are only a finite number of them that are smaller than any given one. One typical application is to show that a given equation has no solutions.

Assuming an example with a particular property exists, one shows that another exists that is in some sense 'smaller' as measured by a natural number. Then by infinitely repeating the same step, one shows there are a yet smaller, then a yet even smaller, example, and hence there must be an infinitude of ever smaller examples. Since there are only a finite number of natural numbers smaller than the size of the initially postulated example, this is impossible — it is a contradiction, so no such initial example can exist.

Proof: Let $\sqrt{k}$ be a non-integer and rational. Then $E = \{b \in \mathbb{N}^* : \exists a(a \in \mathbb{N} \wedge \sqrt{k} = a/b)\} \neq \emptyset$ and by the Well-ordering principle 3.1.1, E contains the smallest value b_1.

Let $\sqrt{k} = a_1/b_1$, $a_1 > 0$, q be the largest positive integer no greater than $\sqrt{k}$, i.e. $\sqrt{k}-1 < q < \sqrt{k}$. Then

$$\sqrt{k} = \frac{a_1}{b_1} = \frac{a_1(\sqrt{k}-q)}{b_1(\sqrt{k}-q)} = \frac{a_1\sqrt{k}-a_1q}{b_1((\frac{a_1}{b_1})-q)} = \frac{b_1\sqrt{k}\times\sqrt{k}-a_1q}{a_1-b_1q} = \frac{b_1k-a_1q}{a_1-b_1q}$$

Let $a_2 = b_1k - a_1q$ and $b_2 = a_1 - b_1q$. Since $q < \sqrt{k}$,

$$a_2 = b_1k - a_1q = a_1(\frac{b_1}{a_1}k - q) = a_1(\frac{k}{\sqrt{k}} - q) = a_1(\sqrt{k}-q) > 0$$
$$b_2 = a_1 - b_1q = b_1(\frac{a_1}{b_1} - q) = b_1(\sqrt{k}-q)$$

This implies $b_2 - b_1 = b_1(\sqrt{k}-q) - b_1 = b_1(\sqrt{k}-q-1) > 0 \Rightarrow b_2 > b_1$.

The positive integer b_2 is smaller than b_1. This contradicts with the definition of b_1.

Remark: It is called "infinite descent" because, we can repeat the process above to get

$$\sqrt{k} = \frac{a_1}{b_1} = \frac{a_2}{b_2} = \frac{a_3}{b_3} = \cdots$$

where $a_3 = b_2k - a_2q$ and $b_3 = a_2 - b_2q$, $\cdots$. The contradiction happens because $b_1 > b_2 > b_3 > \cdots$ but in $\mathbb{N}$, we cannot have a strictly decreasing sequence.

Example 3.2.30. Prove that $\log_2 3$ is rational.

Proof: There are integers m, n such that

$$\log_2 3 = \frac{m}{n}$$

with $n \neq 0$. That would imply that

$$2^{m/n} = 3 \Rightarrow 2^m = 3^n.$$

This implies that an even number equals an odd number. A contradiction.

In general, any $\log_a b$ where $a, b \in \mathbb{Z}$ and a, b are mutually prime is irrational, since $x^a = y^b$ and Unique Factorisation Theorem 3.5.21 leads to a contradiction.

Example 3.2.31. Prove that $\log_{10} 2$ is rational.

Proof.

Example 3.2.32. Use proof by contradiction to show that the sum of any rational number and any irrational number is irrational.

Proof.

Example 3.2.33. Prove that $1 + 3\sqrt{2}$ is irrational.

Proof.

Theorem 3.2.34 (The Infinitude of the Set of Prime Numbers)**.** *The set of prime numbers is infinite.*

Proof: Suppose the set of prime numbers is finite. Then all the prime numbers can be listed, say, in ascending order:

$$p_1 = 2,\ p_2 = 3,\ p_3 = 5,\ p_4 = 7,\ p_5 = 11,\ \cdots,\ p_n.$$

Consider the integer

$$N = p_1 p_2 p_3 \cdots p_n + 1 > 1$$

By Theorem of Divisibility by a Prime (Theorem 3.5.6), N is divisible by some prime number p. Since p is prime, p must equal one of the prime numbers $p_1, p_2, p_3, \cdots, p_n$. Let $p = p_k$ for some $1 \leq k \leq n$ and

$$\begin{aligned} p = p_k &| (N = p_1 p_2 p_3 \cdots p_k \cdots p_n + 1) \\ \Rightarrow & p_k m = p_1 p_2 p_3 \cdots p_k \cdots p_n + 1 \\ \Rightarrow & p_k (m - p_1 p_2 p_3 \cdots p_{k-1} p_{k+1} \cdots p_n) = 1 \end{aligned}$$

for some integer m. This implies 1 can be factorised into two integers which are larger than 1, a contradiction.

§3.3 Disproving Statements

Mathematical statements may not always be true. That is why we need to prove them! Sometimes, we may want to disprove them if we cannot prove them.

Disproving a universal statement may not be difficult if we are able to find a counterexample as illustrated in Section 3.3.1.

Disproving an existential statement would be tougher since we need to prove that its negation, which is a universal statement, is true, as demonstrated in Section 3.3.2.

§3.3.1 Disproving Universal Statements with Counterexamples

Disproving a universal statement

$$\forall x(P(x) \to Q(x)) \tag{3.5}$$

is the same as assuming that

$$\forall x(P(x) \to Q(x)) \to \text{F} \;\Rightarrow \sim \forall x(P(x) \to Q(x)) \equiv \exists x(P(x) \wedge \sim Q(x)).$$

Hence, to disprove (3.5) becomes **finding** a value s, called the **counterexample**, such that $P(s) \wedge \sim Q(s)$ is true.

Example 3.3.1. Disprove the statement "for real numbers n, if n is even, then $\frac{n+2}{2}$ is even" by finding a counterexample.

Solution: Formally, we have $\forall n(\text{even}(n) \to \text{even}(\frac{n+2}{2}))$.

To disprove this statement: Let $n = 4$. 4 is even but $\frac{4+2}{2} = 3$ is not even.

Example 3.3.2. Disprove the following statement by finding a counterexample:

For all real numbers a and b, if $a^2 = b^2$ then $a = b$.

Solution: Formally, the statement can be expressed as

$$\forall a \forall b(a \in \mathbb{R} \wedge b \in \mathbb{R} \to (a^2 = b^2 \to a = b))$$

Though it is of different form from (3.5), to disprove it, we determine its negation

$$\begin{aligned} \sim \forall a \forall b(a \in \mathbb{R} \wedge b \in \mathbb{R} \to (a^2 = b^2 \to a = b)) &\equiv \exists a \exists b \sim (a \in \mathbb{R} \wedge b \in \mathbb{R} \to (a^2 = b^2 \to a = b)) \\ &\equiv \exists a \exists b(a \in \mathbb{R} \wedge b \in \mathbb{R} \wedge \sim (a^2 = b^2 \to a = b)) \\ &\equiv \exists a \exists b(a \in \mathbb{R} \wedge b \in \mathbb{R} \wedge a^2 = b^2 \wedge a \neq b)) \end{aligned}$$

This statement is true, i.e. there are $a = 1$ and $b = -1$ such that $a^2 = 1^2 = 1$, $b^2 = (-1)^2 = 1$ and $a^2 = b^2$ but $a = 1 \neq b = -1$. The original statement is disproved with the counterexample $a = 1$ and $b = -1$.

Example 3.3.3. Disprove the statement "For real numbers a and b, if $a > b$ then $a^2 > b^2$."

Example 3.3.4. Disprove the following statement

For integers m and n, if $2m + n$ is even, then m and n are both even.

Example 3.3.5 (Tutorial 3, Q18)**.** Prove or disprove the following statements:

(a) Every positive integer is the sum of the squares of three integers.

(b) There are 100 consecutive positive integers that are not perfect squares (an integer which can be written as s^2 for some integer s).

Example 3.3.6 (Tutorial 3, Q19)**.** Disprove the following universal statements:

(a) For all real numbers a and b, if $a < b$, then $a^2 < b^2$.

(b) For all integers m and n, if $2m + n$ is odd, then m and n are both odd.

Example 3.3.7 (Tutorial 3, Q21)**.** Determine whether the statement is true or false. Justify your answer with a proof or a counterexample, as appropriate.

(a) The product of any two even integers is even.

(b) For all integers m, if $m > 2$, then $m^2 - 4$ is composite.

(c) For all integers n and m, if $n - m$ is even, then $n^3 - m^3$ is even.

(d) For all integers n, $n^2 - n + 11$ is a prime number.

(e) The quotient of any two rational numbers is a rational number.

(f) If r and s are any two rational numbers, then $\frac{r+s}{2}$ is rational.

§3.3.2 Disproving Existential Statements

Disproving an existential statement

$$\exists x(P(x) \wedge Q(x)) \tag{3.6}$$

is the same as assuming that

$$\exists x(P(x) \wedge Q(x)) \to \text{F} \ \Rightarrow \sim \exists x(P(x) \wedge Q(x)) \equiv \forall x(P(x) \to\sim Q(x)).$$

Hence, to disprove (3.6) becomes **making sure that nothing can make** $P(x) \wedge Q(x)$ true.

Example 3.3.8. Disprove the following quantified statement over the real number domain:

$$\exists x((x \in \mathbb{R}) \wedge (x^2 + x + 1 = 0)).$$

Note: In Topic 1, I regard the $x^2 + x + 1 = 0$ as a predicate, not a statement. Once it is quantified (in this case), it becomes a statement.

Disproof: The negation is true: $\forall x((x \in \mathbb{R}) \to ((x + \frac{1}{2})^2 + \frac{3}{4}) > 0)$.

Example 3.3.9. Show that there is no positive integer n such that $n^2 + 3n + 2$ is prime.

Proof: Let $P(n)$ be the predicate denoting "n is prime". Then

$$\sim \exists n((n > 0) \wedge P((n^2 + 3n + 2) \equiv \forall n((n > 0) \to\sim P((n^2 + 3n + 2)$$

is true since

$$n^2 + 3n + 2 = (n+1)(n+2), \quad n \geq 1$$

is a product of integers $n + 1 > 1$ and $n + 2 > 1$.

By Definition 3.1.8 (**a prime number can only have two factors 1 and itself**), $n^2 + 3n + 2$ has four factors 1, $n + 1$, $n + 2$, $n^2 + 3n + 2$ and is not prime.

Example 3.3.10 (Tutorial 3, Q17)**.** Disprove the existential statement "There exists an integer n such that $6n^2 + 27$ is prime."

Example 3.3.11 (Tutorial 3, Q20)**.** Consider the following existential statement:

There exists an integer x with $x \geq 4$ such that $2x^2 - 5x + 2$ is prime. (*)

1. Give a negation of the statement (*).

2. Prove that the statement (*) is false by showing that its negation is true.

§3.4 Mathematical Induction

Mathematical induction is a method of proof developed to check conjectures about the outcomes of processes that occur repeatedly and according to definite patterns based on the induction principle (page 2).

Let P be a predicate of integer k. Proving a statement

$$\forall n(n \geq a \to P(n))$$

by mathematical induction is a **two-step process**:

1. **Basis Step**: Show that the $P(a)$ is true for a particular integer a.
2. **Inductive Step**: Show that for all integers $k \geq a$, if $P(k)$ is true then $P(k+1)$ is true.

   To perform this step, assume that the property is true for $n = k$ for some integer $k \geq a$. This supposition is called the *inductive hypothesis*. Then show that the property is true for $n = k+1$.

It is based on the following *principle of ordinary mathematical induction.*

Theorem 3.4.1 (Principle of Ordinary Mathematical Induction)**.** *Let $P(n)$ be a predicate that is defined for integers n, and let a be a fixed integer. Suppose the following two statements are true:*

1. *$P(a)$ is true.*
2. *For all integers $k \geq a$, if $P(k)$ is true then $P(k+1)$ is true.*

Then the statement "for all integers $n \geq a$, $P(n)$" is true.

Example 3.4.2 (Induction involving equality)**.** Use mathematical induction to prove that the sum of the first n odd positive integers is n^2 for $n \geq 1$.

Proof:

Let $P(n)$: "$1 + 3 + \cdots + (2n-1) = n^2$".

Basis step: Show that $P(1)$ is true.

$$\text{LHS of } P(1) = 1, \quad \text{RHS of } P(1) = 1^2 = 1 \Rightarrow \text{LHS} = \text{RHS}$$

So $P(1)$ is true.

Inductive step: Suppose that $P(k)$ is true for a positive integer $k \geq 1$, that is, assume

$$1 + 3 + 5 + \cdots + (2k-1) = k^2.$$

and we try to show

$$P(k+1): \; 1 + 3 + 5 + \cdots + (2k-1) + (2(k+1)-1) = (k+1)^2.$$

is true.

$$\begin{aligned}\text{LHS of } P(k+1) &= \underbrace{1 + 3 + 5 + \cdots + (2k-1)}_{\text{using assumption}} + (2k+1) \\ &= k^2 + (2k+1) = (k+1)^2 = \text{RHS of } P(k+1).\end{aligned}$$

Hence $P(k+1)$ is true.

By mathematical induction, $P(n)$ is true for all $n \geq 1$.

Example 3.4.3 (Induction involving equality)**.** Use mathematical induction to prove that

$$\sum_{i=1}^{n} i = 1 + 2 + \cdots + n = \frac{n(n+1)}{2}.$$

Proof.

Example 3.4.4. Use mathematical induction to prove that the equality $\displaystyle\sum_{i=1}^{n} i^3 = \left[\frac{n(n+1)}{2}\right]^2$.

Proof.

We have learn the special cases of binomial theorem, i.e.

$$\begin{aligned}(x+y)^2 &= x^2+2xy+y^2\\ (x+y)^3 &= x^3+3x^2y+3xy^2+y^3\\ (x+y)^4 &= x^4+4x^3y+6x^2y^2+4xy^3+y^4\end{aligned}$$

where the coefficients above are related to **Pascal triangle**.

Theorem 3.4.5 (Binomial Theorem, must memorised). *Let $n \geq 0$ [Epp, 2020, Chapter 9].*

$$(x+y)^n = \sum_{i=0}^{n} \binom{n}{i} x^{n-i}y^i = x^n + \binom{n}{1}x^{n-1}y + \binom{n}{2}x^{n-2}y^2 + \cdots + \binom{n}{n-1}xy^{n-1} + y^n.$$

Proof:

Basis step

$$LHS = (x+y)^0 = 1; \qquad RHS = \binom{0}{0}x^0y^0 = 1$$

Inductive step

Assume that $(x+y)^k = \sum\limits_{i=0}^{k} \binom{k}{i} x^{k-i}y^i$.

$$\begin{aligned}(x+y)^{k+1} &= (x+y)^k \cdot (x+y) = \left(\sum_{i=0}^{k}\binom{k}{i}x^{k-i}y^i\right)\cdot(x+y) = \sum_{i=0}^{k}\binom{k}{i}x^{k-i+1}y^i + \sum_{i=0}^{k}\binom{k}{i}x^{k-i}y^{i+1}\\ &= \binom{k}{0}x^{k+1} + \sum_{\boxed{i=1}}^{\boxed{k}}\binom{k}{i}x^{k-i+1}y^i + \sum_{\boxed{i=0}}^{\boxed{k-1}}\binom{k}{i}x^{k-i}y^{i+1} + \binom{k}{k}y^{k+1}\\ &= x^{k+1} + \sum_{\boxed{i=0}}^{\boxed{k-1}}\binom{k}{i+1}x^{k-(i+1)+1}y^{i+1} + \sum_{i=0}^{k-1}\binom{k}{i}x^{k-i}y^{i+1} + y^{k+1}\\ &= x^{k+1} + \sum_{i=0}^{k-1}\left[\binom{k}{i+1}+\binom{k}{i}\right]x^{k-i}y^{i+1} + y^{k+1}\\ &= \binom{k+1}{0}x^{k+1} + \sum_{i=0}^{k-1}\binom{k+1}{i+1}x^{k-i}y^{i+1} + \binom{k+1}{k+1}y^{k+1}\\ &= \binom{k+1}{0}x^{k+1} + \sum_{i'=1}^{k}\binom{k+1}{i'}x^{k-i'+1}y^{i'} + \binom{k+1}{k+1}y^{k+1} = \sum_{i'=0}^{k+1}\binom{k+1}{i'}x^{k+1-i'}y^{i'}\end{aligned}$$

Note: $\binom{k}{0} = \binom{k}{k} = 1$, $\binom{k+1}{0} = \binom{k+1}{k+1} = 1$ and

$$\begin{aligned}\binom{k}{i+1}+\binom{k}{i} &= \frac{k(k-1)\cdots(k-i)}{(i+1)i(i-1)\cdots 1} + \frac{k(k-1)\cdots(k-i+1)}{i(i-1)\cdots 1} = \frac{k(k-1)\cdots(k-i+1)}{i(i-1)\cdots 1}\left[\frac{k-i}{i+1}+1\right]\\ &= \frac{k(k-1)\cdots(k-i+1)}{i(i-1)\cdots 1}\left[\frac{k+1}{i+1}\right] = \binom{k+1}{i+1}\end{aligned}$$

Example 3.4.6. Use mathematical induction or direct proof to prove that for all integers $n \geq 1$, $2^{2n}-1$ is divisible by 3.

Proof by mathematical induction:

Let $P(n)$: "$3|2^{2n}-1$".

Basis Step: Show that $P(1)$ is true.

$2^{2(1)}-1=3$ is divisible by 3. So $P(1)$ is true.

Inductive Step:

Suppose $P(k)$ is true for a positive integer $k \geq 1$, that is $3|2^{2k}-1$. We must show that $2^{2(k+1)}-1$ is divisible by 3.

$3|2^{2k}-1 \Rightarrow 2^{2k}-1=3a$ for some integer a.

$$\begin{aligned} 2^{2(k+1)}-1=2^{2k+2}-1=4(2^{2k})-1&=3(2^{2k})+(2^{2k}-1)\\ &=3(2^{2k})+3a=3(2^{2k}+a)\Rightarrow 3|2^{2(k+1)}-1. \end{aligned}$$

Thus $P(k+1)$ is true.

Hence, by mathematical induction, $P(n)$ is true for all integers $n \geq 1$.

Proof: By using the Binomial Theorem, we have

$$\begin{aligned} 2^{2n}-1=4^n-1=(3+1)^n-1&=3^n+\binom{n}{1}3^{n-1}+\cdots+\binom{n}{n-1}3+1-1\\ &=3\left[3^{n-1}+\binom{n}{1}3^{n-2}+\cdots+\binom{n}{n-1}\right] \end{aligned}$$

By definition $3|(2^{2n}-1)$.

Example 3.4.7 (Induction with inequality)**.** Use mathematical induction to prove that

$$2n+1<2^n, \quad \text{for all integers } n \geq 3.$$

Proof.

Refer to Tutorial 3 for answers to June 2024 Final Exam Q3 solutions.

§3.5 Divisibility

Divisibility is the basic property of integers. In this section, various concepts and proof techniques associated with divisibility used in Section 3.2 and Section 3.4 are discussed.

Definition 3.5.1. If n and $d \neq 0$ are integers, then n ***is divisible by*** d if there is an integer k such that $n = dk$. In this case, n is called the ***multiple of*** d. d is called the ***factor*** or ***divisor*** of n. We also say that d ***divides*** n and denote it by $d|n$. If d does not divide n, we denote it as $d \nmid n$.

Remark 3.5.2. Based on our definition, *divisors are assumed to be nonzero.* If d is a divisor of n, then n is also divisible by $-d$ (indeed, $n = dk$ implies that $n = (-d)(-k)$), so that the divisors of an integer always occur in pairs. To find all the divisors of a given integer, it is sufficient to obtain the positive divisors and then adjoin to them the corresponding negative integers. For this reason, we shall usually limit ourselves to a consideration of positive divisors.

Example 3.5.3. 1. If a and b are integers, is $4a + 4b$ divisible by 2?

2. Does 4 divides 18?

3. Is 32 a multiple of -16?

4. Is -9 a factor of 54?

5. Suppose a and b are positive integers and $a|b$. Is $a \leq b$?

§3.5.1 Properties of Divisibility

It will be helpful to list some immediate consequences of Definition 3.5.1.

Theorem 3.5.4. *For integers a, b, c, the following hold:*

(a) $a|0$, $1|a$, $a|a$.

(b) $a|1$ *iff* $a = \pm 1$.

(c) If $a|b$ *and* $c|d$*, then* $ac|bd$.

(d) If $a|b$ *and* $b|c$*, then* $a|c$. *(Transitivity of Divisibility)*

(e) $a|b$ *and* $b|a$ *iff* $a = \pm b$.

(f) If $a|b$ *and* $b \neq 0$*, then* $|a| \leq |b|$.

(g) If $a|b$ *and* $a|c$*, then* $a|(bx + cy)$ *for arbitrary integers* x *and* y.

Proof:

(a) $0 = 0(a)$, $a = 1(a)$, $a = a(1)$.

(b) $a|1 \Rightarrow 1 = ak \Rightarrow a = \pm 1$.
$a = \pm 1 \Rightarrow 1 = a^2 \Rightarrow a|1$ by definition.

(c) Suppose $a|b$ and $c|d$, by definition, $b = ak_1$, $d = ck_2$. Hence, $bd = ac(k_1 k_2)$. By definition, $ac|bd$.

(d) Suppose a, b and c are integers such that $a|b$ and $b|c$. Then there are integers r and s such that

$$b = ar, \quad c = bs$$

Hence, $c = (ar)s = a(rs) = ak$, where $k = rs$ is an integer since it is a product of integers r and s. Thus $a|c$.

(e) Suppose $a|b$, $b|a$, by definition, $b = ak_1$ and $a = bk_2$, so $b = bk_2k_1$. Hence $k_2k_1 = 1$ and $k_2 = k_1 = \pm 1$. Therefore, $a = bk_2 = \pm b$.

(f) If $a|b$, then there exists an integer c such that $b = ac$; also, $b \neq 0$ implies that $c \neq 0$. Upon taking absolute values, we get $|b| = |ac| = |a||c|$. Because $c \neq 0$, it follows that $|c| \geq 1$, whence $|b| = |a||c| \geq |a|$.

(g) The relations $a|b$ and $a|c$ ensure that $b = ar$ and $c = as$ for suitable integers r and s. But then whatever the choice of x and y,

$$bx + cy = arx + asy = a(rx + sy).$$

Because $rx + sy$ is an integer, this says that $a|(bx + cy)$, as desired.

Remark 3.5.5. It is worth pointing out that property (g) of Theorem 3.5.4 extends by induction to sums of more than two terms. That is, if $a|b_k$ for $k = 1, 2, ..., n$, then $a|(b_1x_1 + \cdots + b_nx_n)$ for all integers $x_1, ..., x_n$.

Theorem 3.5.6 (Divisibility by a Prime)**.** *Any integer $n > 1$ is divisible by a prime number.*

Proof: Suppose n is a integer that is greater than 1. If n is prime, then n is divisible by a prime number (namely itself), and we are done.

If n is not prime, then

$$n = r_0s_0$$

for some integers $1 < r_0 < n$ and $1 < s_0 < n$. By definition of divisibility, $r_0|n$.

If r_0 is prime, then r_0 is a prime number that divides n, and we are done.

If r_0 is not prime, then

$$r_0 = r_1s_1$$

for some integers $1 < r_1 < r_0$ and $1 < s_1 < r_0$. By definition of divisibility, $r_1|r_0$. But $r_0|n$. Consequently, by transitivity of divisibility (Theorem d), $r_1|n$.

If r_1 is prime, then r_1 is a prime number that divides n, and we are done.

If r_1 is not prime, then

$$r_1 = r_2s_2$$

for some integers $1 < r_2 < r_1$ and $1 < s_2 < r_1$. By definition of divisibility, $r_2|r_1$. But $r_1|n$. Consequently, by transitivity of divisibility (Theorem d), $r_2|n$.

If r_2 is prime, then r_2 is a prime number that divides n, and we are done.

If r_2 is not prime, then we may repeat the above process by factoring r_2 as r_3s_3. Continuing in this way, factoring successive factors of n until we find a prime factor.

We must succeed in a finite number of steps because each new factor is both less than the previous one and greater than 1, and there are fewer than n integers strictly between 1 and n (justified by well-ordering principle). Thus we obtain a sequence r_0, r_1, r_2, $\cdots$, r_k, where $k \geq 0$, $1 < r_k < r_{k-1} < \cdots < r_2 < r_1 < r_0 < n$, and $r_i|n$ for $i = 0, 1, 2, \cdots, k$. The condition for termination is that r_k should be prime. Hence r_k is a prime number that divided n.

§3.5.2 Greatest Common Divisor

An integer d is said to be a ***common divisor/factor of two integers*** a ***and*** b if $d|a$ and $d|b$. Because 1 is a divisor of every integer, 1 is a common divisor of a and b; hence, the set of positive common divisors for a and b is nonempty.

The most important common divisor is the greatest common divisor/factor (abbreviated as $\gcd(a,b)$) which has applications in modular arithmetic (Section 3.6) and thus encryption algorithms such as RSA (Section 3.8.4).

Definition 3.5.7. Let a and b be given integers, with at least one of them different from zero. The ***greatest common divisor* of** a **and** b, denoted $\gcd(a,b)$, is an integer d with the following properties:

1. d is a common divisor of both a and b, i.e. $d|a$ and $d|b$.
2. For all integers c, if $c|a$ and $c|b$, then $c \leq d$.

Example 3.5.8. Determine the greatest common divisor of -12 and 30.

Solution:

Positive divisors of $-12 = \{\ 1,\ 2,\ 3,\ 4,\ 6,\ 12\ \}$
Positive divisors of $30 = \{\ 1,\ 2,\ 3,\ 5,\ 6,\ 10,\ 15,\ 30\ \}$
Positive common divisors of -12 and 30 are $\{\ 1,\ 2,\ 3,\ 6\ \}$.
Because 6 is the largest of these integers, $\gcd(-12,30) = 6$ by definition.

Example 3.5.9. $\gcd(-5,5) = 5$, $\gcd(8,17) = 1$, $\gcd(-8,-36) = 4$.

The next theorem indicates that $\gcd(a,b)$ can be represented as a linear combination of a and b.

Theorem 3.5.10. *Given integers a and b, not both of which are zero, there exist integers x and y such that*

$$\gcd(a,b) = ax + by.$$

Proof: Consider the set S of all positive linear combinations of a and b:

$$S = \{au + bv | au + bv > 0;\ u, v \text{ integers}\}$$

Notice first that $S \neq \emptyset$. For example, if $a \neq 0$, then the integer $|a| = au + b \cdot 0$ lies in S, where we choose $u = 1$ or $u = -1$ according as a is positive or negative.

By virtue of the Well-Ordering Principle, S must contain a smallest element d. Thus, from the very definition of S, there exist integers x and y for which $d = ax + by$.

We claim that $d = \gcd(a,b)$.

Taking stock of the Division Algorithm, we can obtain integers q and r such that $a = qd + r$, where $0 \leq r < d$. Then r can be written in the form

$$r = a - qd = a - q(ax + by) = a(1 - qx) + b(-qy).$$

If r were positive, then this representation would imply that r is a member of S, contradicting the fact that d is the least integer in S (recall that $r < d$). Therefore, $r = 0$, and so $a = qd$, or equivalently $d|a$.

By similar reasoning, $d|b$, the effect of which is to make d a common divisor of a and b.

Now if c is an arbitrary positive common divisor of the integers a and b, then part (g) of Theorem 3.5.4 allows us to conclude that $c|(ax + by)$; that is, $c|d$. By part (f) of the same theorem,

$c = |c| \leq |d| = d$, so that d is greater than every positive common divisor of a and b. Piecing the bits of information together, we see that $d = \gcd(a, b)$ by Definition 3.5.7.

A perusal of the proof of Theorem 3.5.10 reveals that the greatest common divisor of a and b may be described as the smallest positive integer of the form $ax + by$.

Example 3.5.11. Consider the case in which $a = 6$ and $b = 15$. Here, the set S becomes

$$S = \{6u + 15v | u, v \in \mathbb{Z}\} = \{6(-2) + 15 \cdot 1, 6(-1) + 15 \cdot 1, 6 \cdot 1 + 15 \cdot 0, \cdots\} = \{3, 9, 6, \cdots\}.$$

We observe that 3 is the smallest integer in S, whence $3 = \gcd(6, 15)$.

The nature of the members of S appearing in this illustration suggests another result, which we give in the next corollary.

Corollary 3.5.12. *If a and b are given integers, not both zero, then the set $T = \{ax + by | x, y \in \mathbb{Z}\}$ is precisely the set of all multiples of $d = \gcd(a, b)$.*

Proof: $\Rightarrow$: Because $d|a$ and $d|b$, we know that $d|(ax + by)$ for all integers x, y. Thus, every member of T is a multiple of d.

$\Leftarrow$: Conversely, d may be written as $d = ax_0 + by_0$ for suitable integers x_0 and y_0, so that any multiple nd of d is of the form

$$nd = n(ax_0 + by_0) = a(nx_0) + b(ny_0).$$

Hence, nd is a linear combination of a and b, and, by definition, lies in T.

It may happen that 1 and -1 are the only common divisors of a given pair of integers a and b, whence $\gcd(a, b) = 1$. E.g. $\gcd(2, 5) = \gcd(-9, 16) = \gcd(-27, -35) = 1$. This situation occurs often enough to prompt a definition.

Definition 3.5.13. Integers a and b, not both of which are zero, are called ***relatively prime*** or ***mutually prime*** or ***coprime*** if $\gcd(a, b) = 1$. Integers $a_1, a_2, a_3, \cdots, a_n$ are ***(pairwise) relatively prime*** if $\gcd(a_i, a_j) = 1$ for all integers i and j with $1 \leq i, j \leq n$, and $i \neq j$.

The following theorem characterises relatively prime integers in terms of linear combinations.

Theorem 3.5.14. *Let a and b be integers, not both zero. Then a and b are relatively prime iff there exist integers x and y such that $1 = ax + by$.*

Proof:

$\Rightarrow$: If a and b are relatively prime so that $\gcd(a, b) = 1$, then Theorem 3.5.10 guarantees the existence of integers x and y satisfying $1 = ax + by$.

$\Leftarrow$: As for the converse, suppose that $1 = ax+by$ for some choice of x and y, and that $d = \gcd(a, b)$. Because $d|a$ and $d|b$, Theorem 3.5.4 yields $d|(ax + by)$, or $d|1$. Inasmuch as d is a positive integer, this last divisibility condition forces d to equal 1 (part (b) of Theorem 3.5.4 plays a role here), and the desired conclusion follows.

This result leads to an observation that is useful in certain situations; namely,

Corollary 3.5.15. *If* $\gcd(a, b) = d$, *then* $\gcd(a/d, b/d) = 1$.

Proof: The fractions a/d and b/d are integers because d is a divisor both of a and of b.

Now, knowing that $\gcd(a, b) = d$, it is possible to find integers x and y such that $d = ax + by$. Upon dividing each side of this equation by d, we obtain the expression

$$1 = (\frac{a}{d})x + (\frac{b}{d})y.$$

Because a/d and b/d are integers, an appeal to the theorem is legitimate. The conclusion is that a/d and b/d are relatively prime.

Example 3.5.16. As an illustration of this corollary, observe that

$$\gcd(-12, 30) = 6 \Rightarrow \gcd(-12/6, 30/6) = \gcd(-2, 5) = 1$$

Corollary 3.5.17. *If $a|c$ and $b|c$, with $\gcd(a, b) = 1$, then $ab|c$.*

Proof: Inasmuch as $a|c$ and $b|c$, integers r and s can be found such that $c = ar = bs$. Now the relation $\gcd(a, b) = 1$ allows us to write $1 = ax + by$ for some choice of integers x and y. Multiplying the last equation by c, it appears that

$c = c \cdot 1 = c(ax + by) = acx + bcy.$

If the appropriate substitutions are now made on the right-hand side, then

$c = a(bs)x + b(ar)y = ab(sx + ry)$ or, as a divisibility statement, $ab|c$.

Our next result seems mild enough, but is of fundamental importance.

Theorem 3.5.18 (Euclid's lemma)**.** *If $a|bc$, with $\gcd(a, b) = 1$, then $a|c$.*

Proof: From Theorem 3.5.10, writing $1 = ax + by$, where x and y are integers. Multiplication of this equation by c produces

$$c = 1 \cdot c = (ax + by)c = acx + bcy.$$

Because $a|ac$ and $a|bc$, it follows that $a|(acx + bcy)$, which can be recast as $a|c$.

The subsequent theorem often serves as a definition of $\gcd(a, b)$. The advantage of using it as a definition is that order relationship is not involved. Thus, it may be used in algebraic systems having no order relation.

Theorem 3.5.19. *Let a, b be integers, not both zero. For a positive integer d, $d = \gcd(a, b)$ iff*

(a) $d|a$ and $d|b$.

(b) Whenever $c|a$ and $c|b$, then $c|d$.

Proof:

$\Rightarrow$: Suppose that $d = \gcd(a, b)$. Certainly, $d|a$ and $d|b$, so that (a) holds. In light of Theorem 3.5.10, d is expressible as $d = ax + by$ for some integers x, y. Thus, if $c|a$ and $c|b$, then $c|(ax + by)$, or rather $c|d$. In short, condition (b) holds.

$\Leftarrow$: Let d be any positive integer satisfying the stated conditions. Given any common divisor c of a and b, we have $c|d$ from hypothesis (b). The implication is that $d \geq c$, and consequently d is the greatest common divisor of a and b.

§3.5.3 Unique Factorisation Theorem & Standard Factor Form

Prime numbers are special factors because they are the only factors greater than 1. Hence, a natural question arises, how many prime numbers are there? Can integer greater than one be written as a product of prime(s)?

Proposition 3.5.20. *If p is a prime and $a, b \in \mathbb{Z}$ such that $p|ab$, then $p|a$ or $p|b$.*

Proof: Assume $p \nmid a$. Then $\gcd(a, p) = 1$. By Euclid's Lemma 3.5.18, $p|b$. □

Any integer > 1 is either prime or can be written as a product of primes. The following theorem characterises this property.

Theorem 3.5.21 (**Unique Factorisation Theorem** for the Integers)**.** *Given any integer $n > 1$, there is a positive integer k, distinct prime numbers $p_1, p_2, \cdots, p_k$, and there are positive integers e_1, e_2, $\cdots$, e_k such that*

$$n = p_1^{e_1} p_2^{e_2} p_3^{e_3} \cdots p_k^{e_k},$$

and any other expression of n as a product of prime numbers is identical to this except, perhaps, for the order in which the factors are written.

Definition 3.5.22. Given any integer $n > 1$, the ***standard factored form*** of n is an expression of the form

$$n = p_1^{e_1} p_2^{e_2} p_3^{e_3} \cdots p_k^{e_k}, \quad p_1 < p_2 < \cdots < p_k,$$

where k is a positive integer; $p_1, p_2, \cdots, p_k$ are prime numbers; $e_1, e_2, \cdots, e_k$ are positive integers.

Standard Factored Form in Standard ML

```
(* https://wiki.haskell.org/99_questions/Solutions/35 *)
fun primeFactors n =
  let
    fun primeFactorsAux n f = if f*f > n then
            [n] (* n is a prime if it does not have smaller factors *)
          else if (n mod f = 0) then
            [f] @ (primeFactorsAux (n div f) f)
          else
            primeFactorsAux n (f + 1)
  in
    primeFactorsAux n 2
  end;

fun printIntList x =
  let
    fun join (s : int list) = (* Taken from topic1ttbl.sml *)
      case s of
        [] => ""
      | x::xs => if xs=[] then (Int.toString x) else (Int.toString x)^" * "^(join xs)
  in
    print (join x ^ "\n")
  end

val _ = printIntList (primeFactors 28234423783);
```

Example 3.5.23. Suppose m is an integer such that

$$8 \cdot 7 \cdot 6 \cdot 5 \cdot 4 \cdot 3 \cdot 2 \cdot m = 17 \cdot 16 \cdot 15 \cdot 14 \cdot 13 \cdot 12 \cdot 11 \cdot 10.$$

Does $17|m$?

Example 3.5.24. In Topic 1, Example 1.1.1, write 28234423783 in standard factored form.

Solution: Using the Standard ML program, we obtain a standard factored form

$$28234423783 = 229 \ * \ 641 \ * \ 192347$$

Example 3.5.25. Write 3300 in standard factored form.

§3.5.4 Quotient-Remainder Theorem, div and mod

Theorem 3.5.26 (Quotient-Remainder Theorem). *Given any integer n and positive integer d, there exist unique integers q and r such that*

$$n = qd + r, \quad 0 \leq r < d.$$

Definition 3.5.27. With the notation of Theorem 3.5.26, r is called the ***remainder*** of the division of n by d. If $r = 0$, we say that n is a ***multiple*** of d, or that n is ***divisible by*** d, or d **is a *divisor of*** n, or that d ***divides*** n, or that d **is a *factor of*** n. The number q is called the ***quotient*** of n by d and is denoted n div d.

We can also define the modulo operator:

$$n \text{ mod } d = r.$$

Example 3.5.28. For each values of n and d, find integers q and r such that $n = dq + r$ and $0 \leq r < d$.

	n	d	q	r	q, r with Standard ML
(i)	54	4			`54 div  4, 54 mod  4`
(ii)	-54	4			`~54 div  4, ~54 mod  4`
(iii)	54	70			`54 div 70, 54 mod 70`
(iv)	-54	70			`~54 div 70, ~54 mod 70`
(v)	32	9			`32 div  9, 32 mod  9`

Example 3.5.29. What day of the week will it be 1 year from today?

§3.5.5 Floor and Ceiling

It is often convenient to transform the problems of divisibility from the set of integers $\mathbb{Z}$ to the set of real numbers $\mathbb{R}$ and vice versa. The two operations, floor and ceiling, that relates $\mathbb{Z}$ and $\mathbb{R}$ are defined below.

Definition 3.5.30. Given any real number x, the *floor* of x, denoted $\lfloor x \rfloor$, is defined as follows:

$$\lfloor x \rfloor := \text{the unique integer } n \text{ such that } n \leq x < n+1.$$

Symbolically,

$$\lfloor x \rfloor = n, \quad n \leq x < n+1.$$

Definition 3.5.31. Given any real number x, the *ceiling* of x, denoted $\lceil x \rceil$, is defined as follows:

$$\lceil x \rceil := \text{the unique integer } n \text{ such that } n-1 < x \leq n.$$

Symbolically,

$$\lceil x \rceil = n, \quad n-1 < x \leq n.$$

Example 3.5.32. Compute $\lfloor x \rfloor$ and $\lceil x \rceil$ for each of the following values of x.

	x	$\lfloor x \rfloor$	$\lceil x \rceil$	Standard ML
(i)	25/4			`floor (25.0/4.0); ceil (25.0/4.0);`
(ii)	0.999			`floor (0.999); ceil (0.999);`
(iii)	$0.999\cdots$			`floor (1.0); ceil (1.0);`
(iv)	-2.01			`floor (~2.01); ceil (~2.01);`
(v)	$\lfloor -\frac{1}{2} \rfloor + \lfloor \frac{2}{3} \rfloor$			already in integer form, not supported

Example 3.5.33. The 1,370 soldiers at a military base are given the opportunity to take buses into town for an evening out. Each bus holds a maximum of 40 passengers.

1. For reasons of economy, the base commander will send only full buses. What is the maximum number of buses the base commander will send?

2. If the base commander is willing to send a partially filled bus, how many buses will the commander need to allow all the soldiers to take the trip?

Example 3.5.34. If k is an integer, simplify $\lfloor k \rfloor$ and $\lfloor k + \frac{1}{2} \rfloor$ as an expression of k.

Example 3.5.35. Is the following statement true or false?

For all real numbers x and y, $\lfloor x + y \rfloor = \lfloor x \rfloor + \lfloor y \rfloor$.

Theorem 3.5.36. *$\forall x \in \mathbb{R}$, $\forall m \in \mathbb{Z}$, $\lfloor x+m \rfloor = \lfloor x \rfloor + m$.*

Proof: For any given real number x and any integer m. Let $n = \lfloor x \rfloor$. By definition of floor, n is an integer and $n \leq x < n+1$. Add m to all sides gives

$$n+m \leq x+m < n+m+1.$$

Now $n+m$ is an integer, and so, by definition of floor, the left-hand side of the equation to be shown is

$$\lfloor x+m \rfloor = n+m = \lfloor x \rfloor + m.$$

□

Theorem 3.5.37. *For any integer n,*

$$\left\lfloor \frac{n}{2} \right\rfloor = \begin{cases} \frac{n}{2}, & \text{if } n \text{ is even,} \\ \frac{n-1}{2}, & \text{if } n \text{ is odd.} \end{cases}$$

Proof: When n is odd, $n = 2k+1$ for some integer k and

$$\left\lfloor \frac{n}{2} \right\rfloor = \left\lfloor \frac{2k+1}{2} \right\rfloor = \left\lfloor k + \frac{1}{2} \right\rfloor = k + \left\lfloor \frac{1}{2} \right\rfloor = k + 0 = k = \frac{n-1}{2}$$

using Theorem 3.5.36.

When n is even, $n = 2k$ for some integer k and

$$\left\lfloor \frac{n}{2} \right\rfloor = \left\lfloor \frac{2k}{2} \right\rfloor = \lfloor k \rfloor = k = \frac{n}{2}$$

□

Theorem 3.5.38 (Raymond T. Boute)**.** *If n is an integer and d is a nonzero integer, and if $q = \lfloor \frac{n}{|d|} \rfloor$ and $r = n - d\lfloor \frac{n}{|d|} \rfloor$, then*

$$n = dq + r, \quad 0 \leq r < d.$$

Proof: Suppose n is a nonnegative integer, d is a positive integer, $q = \lfloor \frac{n}{d} \rfloor$ and $r = n - d\lfloor \frac{n}{d} \rfloor$. Then

$$dq + r = d\left\lfloor \frac{n}{d} \right\rfloor + \left(n - d\left\lfloor \frac{n}{d} \right\rfloor\right) = n$$

and

$$q = \left\lfloor \frac{n}{d} \right\rfloor \Rightarrow q \leq \frac{n}{d} < q+1 \Rightarrow dq \leq n < dq + d \Rightarrow 0 \leq n - dq < d.$$

But $n - dq = r$. Hence $0 \leq r < d$. □

Remark 3.5.39. From this we have n div $d = \lfloor \dfrac{n}{d} \rfloor$ and $n \mod d = n - |d| \lfloor \dfrac{n}{|d|} \rfloor$.

§3.6 Modular Arithmetic

> Using the modular arithmetic, we can define "congruence", an equivalence relation over $\mathbb{Z}$ (Topic 4). Modular arithmetic also has applications in cryptography (Section 3.8) and random number generation (Section 3.9).

§3.6.1 Properties of Congruence

Definition 3.6.1. Let $a,\ b \in \mathbb{Z}$ and $n \in \mathbb{Z}^+$. We say a and b are ***congruent modulo*** n provided that $n|(a-b)$. We write $a \equiv b \pmod{n}$ or $a \equiv_n b$ which means $a - b = kn$ for some integer k.

When $n \nmid (a-b)$, we say that a **is *incongruent to*** b ***modulo*** n, and in this case we write $a \not\equiv b \pmod{n}$.

Remark 3.6.2. Note that $\equiv$ for number theory is different from the logical equivalence found in mathematical logic which is used in Topics 1 and 2.

Example 3.6.3. For $n = 7$,

$$3 \equiv 24 \pmod{7},\ -31 \equiv 11 \pmod{7},\ -15 \equiv -64 \pmod{7}$$

because $3 - 24 = (-3)7$, $-31 - 11 = (-6)7$, and $-15 - (-64) = 7 \cdot 7$.

$25 \not\equiv 12 \pmod{7}$, because 7 fails to divide $25 - 12 = 13$.

Example 3.6.4. 1. $12 \equiv 7 \pmod{5}$? ....

2. $-6 \equiv -8 \pmod{4}$? ....

3. $0 \equiv -6 \pmod{3}$? ....

Theorem 3.6.5. *Let a, b, and $n > 1$ be any integers. The following statements are all equivalent:*

1. $n|(a-b)$
2. $a \equiv b \pmod{n}$
3. $a = b + kn$ *for some integer* k
4. a *and* b *have the same (nonnegative) remainder when divided by* n
5. $a \bmod n = b \bmod n$

Remark 3.6.6.
- Two integers are congruent modulo 2 when they are both even or both odd
- Any two integers are congruent modulo 1
- Congruence modulo 1 is not particularly interesting, the usual practice is to assume that $n > 1$.

Given an integer a, let q and r be its quotient and remainder upon division by n, so that

$$a = qn + r, \quad 0 \le r < n.$$

Then, by definition of congruence, $a \equiv r \pmod{n}$. Because there are n choices for r, we see that every integer is congruent modulo n to exactly one of the values $0, 1, 2, ..., n-1$; in particular, $a \equiv 0 \pmod{n}$ iff $n|a$. The set of n integers 0, 1, 2,..., $n-1$ is called the set of ***least nonnegative residues modulo*** n.

Theorem 3.6.7. *$\forall a, b \in \mathbb{Z}$, $a \equiv b \pmod{n}$ iff a and b leave the same nonnegative remainder when divided by n.*

Theorem 3.6.8 (Basic Arithmetic of Congruences)**.** *Let $n > 1$ be fixed and a, b, c, d be arbitrary integers. Then the following properties hold:*

(a) $a \equiv a \pmod{n}$*.*
(b) If $a \equiv b \pmod{n}$*, then* $b \equiv a \pmod{n}$*.*
(c) If $a \equiv b \pmod{n}$ *and* $b \equiv c \pmod{n}$*, then* $a \equiv c \pmod{n}$*.*
(d) If $a \equiv b \pmod{n}$ *and* $c \equiv d \pmod{n}$*, then* $a + c \equiv b + d \pmod{n}$ *and* $ac \equiv bd \pmod{n}$*.*
(e) If $a \equiv b \pmod{n}$*, then* $a + c \equiv b + c \pmod{n}$ *and* $ac \equiv bc \pmod{n}$*.*
(f) If $a \equiv b \pmod{n}$*, then* $a^k \equiv b^k \pmod{n}$ *for any positive integer* k*.*

Proof:

(a) For any integer a, we have $a - a = 0 \cdot n$, so that $a \equiv a \pmod{n}$.

(b) If $a \equiv b \pmod{n}$, then $a - b = kn$ for some integer k. Hence, $b - a = -(kn) = (-k)n$ and because $-k$ is an integer, this yields property (b).

(c) Suppose that $a = b \pmod{n}$ and also $b \equiv c \pmod{n}$. Then there exist integers h and k satisfying $a - b = hn$ and $b - c = kn$. It follows that

$$a - c = (a - b) + (b - c) = hn + kn = (h + k)n$$

which is $a \equiv c \pmod{n}$ in congruence notation.

(d) Similar to (c), if $a \equiv b \pmod{n}$ and $c \equiv d \pmod{n}$, then we are assured that $a - b = k_1 n$ and $c - d = k_2 n$ for some choice of k_1 and k_2. Adding these equations, we obtain

$$(a + c) - (b + d) = (a - b) + (c - d) = k_1 n + k_2 n = (k_1 + k_2)n$$

or, as a congruence statement, $a + c \equiv b + d \pmod{n}$.
As regards the second assertion of property (d), note that

$$ac = (b + k_1 n)(d + k_2 n) = bd + (bk_2 + dk_1 + k_1 k_2 n)n$$

Because $bk_2 + dk_1 + k_1 k_2 n$ is an integer, this says that $ac - bd$ is divisible by n, whence $ac = bd \pmod{n}$.

(e) The proof of property (e) is covered by (d) and the fact that $c \equiv c \pmod{n}$.

(f) The statement certainly holds for $k = 1$, and we will assume it is true for some fixed k. From (d), we know that $a \equiv b \pmod{n}$ and $a^k \equiv b^k \pmod{n}$ together imply that $aa^k \equiv bb^k \pmod{n}$, or equivalently $a^{k+1} = b^{k+1} \pmod{n}$. Hence, the induction step is complete.

Corollary 3.6.9 (**Congruence** n **is an Equivalence Relation**)**.** *The congruence relation* $\equiv_n$ *is an equivalence relation on* $\mathbb{Z}$ *and the map* $\{0, 1, \cdots, n-1\} \to \mathbb{Z}/\equiv_n, \quad r \mapsto \bar{r} = r + n\mathbb{Z}$ *is a bijection.*

Proof: From the Basic Arithmetic Theorem 3.6.8 (a), (b) and (c), we know that $\equiv_n$ is reflexive, symmetric and transitive. Therefore $\equiv_n$ is an equivalence relation.

The map is well-defined since for every $r = 0, 1, \cdots, n-1$, there is a set $r + n\mathbb{Z}$ corresponding to it.

To show that it is bijection, we show that it is an injection: assume $\bar{r} = \bar{s}$ with $0 \leq r, s < n$. Then, by definition, $r \equiv s \pmod{n}$, so $n | r - s$ and $|r - s| < n$, therefore $r = s$.

To show that the map is a surjection: Let $\bar{a} \in \mathbb{Z}/\equiv_n$, by definition, $\bar{a} = \{a + nk : k \in \mathbb{Z}\}$, by Quotient-Remainder Theorem, there is an $r \in \{0, 1, \cdots, n-1\}$ such that $a = r + nm$, hence $r \equiv_n a$ and by definition, $\bar{r} = \bar{a}$.

In mathematics, $\mathbb{Z}/\equiv_n$ is denoted $\mathbb{Z}/n\mathbb{Z}$.

Modular arithmetic is important because all sorts of questions that are difficult to answer with respect to $\mathbb{Z}$ are effectively (though not necessarily efficiently, if n is large) decidable with respect to $\mathbb{Z}/n\mathbb{Z}$. If we can show in this way that something is impossible over $\mathbb{Z}/n\mathbb{Z}$, then this often implies a negative answer for $\mathbb{Z}$, too.

Some of the elementary properties of equality (reflexive, symmetry and transitivity) will make multiplication simple with congruence.

Corollary 3.6.10. *Let a, b, and n be integers with $n > 1$. Then*

$$ab \equiv [(a \bmod n)(b \bmod n)] \pmod n$$

or, equivalently,

$$ab \bmod n = [(a \bmod n)(b \bmod n)] \bmod n$$

In particular, if m is a positive integer, then

$$am \equiv [(a \bmod n)m] \pmod n.$$

***Note**: Differentiate between the congruence relation* $\pmod n$ *and the operation "mod n".*

Example 3.6.11. Show that 41 divides $2^{20} - 1$.

Proof: We begin by noting that $2^5 \equiv -9 \pmod{41}$, whence $(2^5)^4 \equiv (-9)^4 \pmod{41}$ by Theorem 3.6.8(f); in other words, $2^{20} \equiv 81 \cdot 81 \pmod{41}$. But $81 \equiv -1 \pmod n$, and so $81 \cdot 81 \equiv 1 \pmod{41}$. Using parts (b) and (e) of Theorem 3.6.8, we finally arrive at

$$2^{20} - 1 \equiv 81 \cdot 81 - 1 \equiv 1 - 1 \equiv 0 \pmod{41}$$

Thus, $41 | 2^{20} - 1$, as desired.

Example 3.6.12. Calculate $12^{43} \bmod 713$.

Solution: When the power is larger, we need to get the "binary" decomposition of the power:

$$43 = 32 + 8 + 2 + 1 = 2^5 + 2^3 + 2 + 1$$

so that we have

$$\begin{aligned} 12^{43} \bmod 713 &= [(12^{32})(12^8)(12^2)(12)] \bmod 713 \\ &= [(12^{32} \bmod 713)(12^8 \bmod 713)(12^2 \bmod 713)(12 \bmod 713)] \bmod 713 \\ &= (485)(629)(144)(12) \bmod 713 = 527152320 \bmod 713 = 48 \end{aligned}$$

where

$$\begin{aligned} 12^2 \bmod 713 &= 144 \\ 12^8 \bmod 713 &= 144^4 \bmod 713 = 629 \\ 12^{16} \bmod 713 &= (692)^2 \bmod 713 = 639 \\ 12^{32} \bmod 713 &= (639)^2 \bmod 713 = 485 \end{aligned}$$

The method mentioned above is not how real-world algorithm performs calculations.

An implementation of modular exponentiation in Standard ML

```
fun powMod a n m = let
    fun f _ 0 acc = acc
      | f x e acc = let
          val x2 = x*x mod m
          val q  = e div 2
          val m2 = if (e mod 2 = 1) then (x*acc) mod m else acc
        in (
          print ("x^2="^(Int.toString x2)^"; q="^(Int.toString q)^"; ");
          print ("m2="^(Int.toString m2)^"\n");
          f x2 q m2 )
        end
  in
    (f ((abs a) mod m) n 1) mod m
  end

(* Example 3.6.12 *)
val _ = print (Int.toString (powMod 12 43 713) ^ "\n")
```

Example 3.6.13. Work through Example 3.6.12 using `powMod` algorithm.
Solution: The initial values $x = 12$, $q = 43$, m2=1.

x^2	$q/2$	$q \bmod 2$	m2
$12^2 = 144 \equiv_{713} 144$	43/2=21	1	12
$144^2 = 20736 \equiv_{713} 59$	21/2=10	1	$12 \times 144 = 1728 \equiv_{713} 302$
$59^2 = 3481 \equiv_{713} 629$	10/2=5	0	302
$629^2 = 595641 \equiv_{713} 639$	5/2=2	1	$302 \times 629 = 189958 \equiv_{713} 300$

Example 3.6.14. Calculate $144^4 \bmod 713$

Theorem 3.6.15 (Fermat's Little Theorem)**.** *If p is any prime number and a is any integer, then $a^p \equiv a \pmod{p}$. If $p \nmid a$, then $a^{p-1} \equiv 1 \pmod{p}$.*

Example 3.6.16. Calculate $7^{11^{13}} \bmod 17$.

§3.6.2 Euclidean Algorithm

The ***Euclidean algorithm*** (also called *Euclid's algorithm*) is an efficient method for computing the greatest common divisor of two integers. It is named after the Greek mathematician Euclid, who described it in Books VII and X of his Elements (`https://en.wikipedia.org/wiki/Euclid%27s_Elements`).

The algorithm has many theoretical and practical applications. It is a key element of the RSA algorithm (Section 3.8.4). It is used to solve Diophantine equations, such as finding numbers that satisfy multiple congruences (Chinese remainder theorem) or multiplicative inverses of a finite field. It can also be used to construct continued fractions, in the Sturm chain method for finding real roots of a polynomial, and in several modern integer factorisation algorithms. Finally, it is a basic tool for proving theorems in modern number theory, such as Lagrange's four-square theorem and the fundamental theorem of arithmetic (unique factorisation).

The Euclidean algorithm is based on the following two properties: Let $a, b, q, r \in \mathbb{Z}$

- If $a > 0$ then $\gcd(a, 0) = a$.
- If $b \neq 0$, $q \geq 0$, $r \geq 0$, $a = bq + r$ then $\gcd(a, b) = \gcd(b, r)$

An implementation of Euclidean algorithm in Standard ML

```
(* Euclidean Algorithm for GCD *)
fun gcd (a, 0) = a
  | gcd (a, b) = gcd (b, a mod b)
```

Example 3.6.17. Use the Euclidean algorithm to find $\gcd(330, 156)$.

Solution:

$$\begin{aligned}
\gcd(330,156) &= \gcd(156,18) && [330 = 156(2) + 18] \\
&= \gcd(18,12) && [156 = 18(8) + 12] \\
&= \gcd(12,6) && [18 = 12(1) + 6] \\
&= \gcd(6,0) && [12 = 6(2) + 0] \\
&= 6
\end{aligned}$$

Definition 3.6.18. An integer d is said to be a *linear combination of integers a and b* if there exist integers s and t such that $as + bt = d$.

Theorem 3.6.19. *For all integers a and b, not both zero, if $d = \gcd(a, b)$, then there exist integers s and t such that $as + bt = d$.*

Example 3.6.20. Express $\gcd(330, 156)$ as a linear combination of 330 and 156.

Solution: From Example 3.6.17,

$$\begin{aligned}
6 &= 18 - 12 \\
&= 18 - [156 - 8(18)] \\
&= 18 + (-1)(156) + 8(18) \\
&= 9(18) + (-1)(156) \\
&= 9[330 - 2(156)] + (-1)(156) \\
&= 9(330) + (-18)(156) + (-1)(156) \\
&= 9(330) + (-19)(156)
\end{aligned}$$

Hence $\gcd(330, 156) = 9(330) + (-19)(156)$.

Example 3.6.21. Show that 660 and 43 are relatively prime, and prove that the statement $\exists u \exists v(660u + 43v = 1)$ is true in the **integer** domain.

Theorem 3.6.22. *If $ca \equiv cb \pmod{n}$, then $a \equiv b \pmod{n/d}$, where $d = \gcd(c, n)$.*

Theorem 3.6.22 gets its maximum force when the requirement that $\gcd(c, n) = 1$ is added, for then the cancellation may be accomplished without a change in modulus.

Corollary 3.6.23. *If $ca \equiv cb \pmod{n}$ and $\gcd(c, n) = 1$, then $a \equiv b \pmod{n}$.*

Corollary 3.6.24. *If $ca \equiv cb \pmod{p}$ and $p \nmid c$, where p is prime, then $a \equiv b \pmod{p}$.*

Example 3.6.25. $33 \equiv 15 \pmod{9} \Rightarrow 3 \cdot 11 \equiv 3 \cdot 5 \pmod{9} \Rightarrow 11 \equiv 5 \pmod{3}$

Corollary 3.6.26. *For all integers a and n, if $\gcd(a, n) = 1$, then there exists an integers s such that $as \equiv 1 \pmod{n}$. The integer s is called the* inverse of a modulo n.

Example 3.6.27.

1. Find an inverse for 43 modulo 660. That is, find an integer s such that $43s \equiv 1 \pmod{660}$.

2. Find a positive inverse for 3 modulo 40. That is, find a positive integer s such that $3s \equiv 1 \pmod{40}$.

§3.6.3 Linear Congruences

Solving a general equation in modular arithmetic is impossible. For a linear equation

$$ax \equiv b \pmod{n} \tag{3.7}$$

which is is called a ***linear congruence***, the theory of linear congruences allows us to find a solution $x_0 \in \mathbb{Z}$ for which $ax_0 \equiv b \pmod{n}$.

By definition, $ax_0 \equiv b \pmod{n}$ iff $n|ax_0 - b$ or $ax_0 - b = ny_0$ for some integer y_0. Thus, the problem of finding all integers that will satisfy the linear congruence (3.7) is identical with that of obtaining all solutions of the **linear Diophantine equation**

$$ax - ny = b. \tag{3.8}$$

Theorem 3.6.28. *The linear congruence* (3.7) *has a solution iff $d|b$, where $d = \gcd(a, n)$. If $d|b$, then it has d mutually incongruent solutions modulo n.*

Proof: Note that the given congruence is equivalent to the linear Diophantine equation $ax - ny = b$. From Theorem 2.4.4 of Topic 2, it is known that the latter equation can be solved iff $d|b$; moreover, if it is solvable and x_0, y_0 is one specific solution, then any other solution has the form

$$x = x_0 + \frac{n}{d}t, \quad y = y_0 + \frac{a}{d}t$$

for some choice of t.

Among the various integers satisfying the first of these formulae, consider those that occur when t takes on the successive values $t = 0, 1, 2, ..., d - 1$:

$$x_0, x_0 + \frac{n}{d}, x_0 + \frac{2n}{d}, \cdots, x_0 + \frac{(d-1)n}{d}.$$

We claim that these integers are incongruent modulo n, and all other such integers x are congruent to some one of them. If it happened that

$$x_0 + \frac{n}{d}t_1 \equiv x_0 + \frac{n}{d}t_2 \pmod{n}$$

where $0 \leq t_1 < t_2 \leq d - 1$, then we would have

$$\frac{n}{d}t_1 \equiv \frac{n}{d}t_2 \pmod{n}$$

Now $\gcd(n/d, n) = n/d$, by Theorem 3.6.22 the factor n/d could be cancelled leading to

$$t_1 \equiv t_2 \pmod{d}$$

which is to say that $d|t_2 - t_1$. But this is impossible in view of the inequality $0 < t_2 - t_1 < d$.

It remains to argue that any other solution $x_0 + (n/d)t$ is congruent modulo n to one of the d integers listed above. The Division Algorithm permits us to write t as $t = qd + r$, where $0 \leq r \leq d - 1$. Hence

$$x_0 + \frac{n}{d}t = x_0 + \frac{n}{d}(qd + r) = x_0 + nq + \frac{n}{d}r \equiv x_0 + \frac{n}{d}r \pmod{n}$$

with $x_0 + (n/d)r$ being one of our d selected solutions.

The argument that we gave in Theorem 3.6.28 brings out a point worth stating explicitly: If x_0 is any solution of $ax \equiv b \pmod{n}$, then the $d = \gcd(a, n)$ incongruent solutions are given by

$$x_0, x_0 + \frac{n}{d}, x_0 + 2\frac{n}{d}, \cdots, x_0 + (n-1)\frac{n}{d}.$$

Corollary 3.6.29. *If $\gcd(a, n) = 1$, then the linear congruence $ax \equiv b \pmod{n}$ has a unique solution modulo n.*

Example 3.6.30. Solve the linear congruence $18x \equiv 30 \pmod{42}$.

Solution: Because $\gcd(18, 42) = 6$ and 6 surely divides 30, Theorem 3.6.28 guarantees the existence of exactly six solutions, which are incongruent modulo 42. By inspection, one solution is found to be $x = 4$.

The six solutions are as follows:

$$x \equiv 4 + (42/6)t = 4 + 7t \pmod{42}, \quad t = 0, 1, ..., 5$$

or, plainly enumerated, $x \equiv 4, 11, 18, 25, 32, 39 \pmod{42}$.

Given relatively prime integers a and n, the congruence $ax \equiv 1 \pmod{n}$ has a unique solution. This solution is sometimes called the *(multiplicative) inverse of a modulo n.*

Example 3.6.31. Solve the linear congruence $9x \equiv 21 \pmod{30}$.

To solve a general linear congruence (3.7), we need the `https://en.wikipedia.org/wiki/Chinese_remainder_theorem` which is covered in “Number Theory” course.

§3.7 Application: SMT For Number System

The Satisfiability Modulo Theories (SMT), which is mentioned in Topic 1, can combine the semantic logic from in Topics 1 and 2 and some basic arithmetic of the number systems in this topic to form linear integer arithmetic (LIA) theory and linear integer arithmetic (LRA) theory. In this section, explanations and examples from `https://jfmc.github.io/z3-play/` are adapted.

The basic building blocks of SMT formulas are constants and functions. Constants are just functions that take no arguments. So everything is really just a function.

An Integer Example in Z3 SMT-LIB

```
;;; https://jfmc.github.io/z3-play/
(declare-fun f (Int) Int)
(declare-fun a () Int) ; a is a constant
(declare-const b Int)  ; syntax sugar for (declare-fun b () Int)
(declare-const u Int)
(declare-const v Int)
(assert (> a 20))
(assert (> b a))        ; Are there numbers (a,b) s.t. a>20 and b>a?
(assert (= (f 10) 1))   ; Is there a function f s.t. f(10) = 1?
(assert (and (not (= u 0)) (not (= v 0))))
(assert (= (+ (* 330 u) (* 156 v)) 6))    ; Example 3.6.20
(check-sat)
(get-model)
```

The SMT solver Z3 has builtin support for integer and real constants with support for some linear arithmetic (and extremely limited nonlinear arithmetic). These two types (sorts) represent the mathematical integers and reals rather than machine integers (32-bit or 64-bit) and floating point numbers.

After constants are declared, the user can assert formulas containing these constants. The formulas contain arithmetic operators such as $+$, $-$, $<$, and so on. The command `check-sat` will instruct Z3 to try to **find an interpretation/model** for the declared constants that makes all formulas true. The interpretation is basically assigning a number to each constant. If such interpretation exists, we say it is a model for the asserted formulas. The command `get-model` displays the model built by Z3.

Real constants should contain a decimal point. Unlike most programming languages, Z3 will not convert automatically integers into reals and vice-versa. The function to-real can be used to convert an integer expression into a real one.

Mixed Integer and Real Example in Z3 SMT-LIB

```
;;; https://jfmc.github.io/z3-play/
;;; https://smt-lib.org/examples.shtml
(declare-const a Int)
(declare-const b Int)
(declare-const c Int)
(declare-const d Real)
(declare-const e Real)
(assert (> a b))               ; a > b
(assert (> a (+ b 2)))         ; a > b+2
(assert (= a (+ (* 2 c) 10)))  ; a = 2c+10
(assert (<= (+ c b) 1000))     ; c+b <= 1000
(assert (>= d e))              ; d >= e
(assert (> e (+ (to_real (+ a b)) 2.0)))  ; e > (a+b)+2
(assert (= d (+ (to_real c) 0.5)))        ; d = c + 0.5
(check-sat)
(get-model)
(get-value (a b c d e))
```

Z3 also has support for division, integer division, modulo and remainder operators. Internally, they are all mapped to multiplication.

Mixed Integer and Real Example in Z3 SMT-LIB

```
;;; https://jfmc.github.io/z3-play/
(declare-const a Int)
(declare-const r1 Int)
(declare-const r2 Int)
(declare-const r3 Int)
(declare-const r4 Int)
(declare-const r5 Int)
(declare-const r6 Int)
(assert (= a 10))
(assert (= r1 (div a 4)))     ; integer division
(assert (= r2 (mod a 4)))     ; mod
(assert (= r3 (rem a 4)))     ; remainder
(assert (= r4 (div a (- 4)))) ; integer division
(assert (= r5 (mod a (- 4)))) ; mod
(assert (= r6 (rem a (- 4)))) ; remainder
(declare-const b Real)
(declare-const c Real)
(assert (>= b (/ c 3.0)))
(assert (>= c 20.0))
(check-sat)
(get-model)
```

To illustrate the combination of semantic logic and model theory for number systems, we adopt the Job Shop Problem example from `https://www.cs.toronto.edu/~victorn/tutorials/z3/index.html`

Example 3.7.1 (Job Shop Problem)**.** To schedule tasks for a problem (such as building a bike) on different work stations, with some constraints:

- Each task in a job must start only after the previous task has been completed.
- A task cannot be paused — the time it takes to complete cannot be divided.
- The work stations can only work on one task.

Suppose We have three jobs job0, job1, job2 containing task. We define the problem parameters in the following lists, where each element of the list is a pair (m, d) where m represents the machine where the task has to be executed and d is the duration of the task:

- job0 = [(0,3), (1,2), (2,2)];
- job1 = [(0,2), (2,1), (1,4)];
- job2 = [(1,4),(2,3)];

The jobs are a list of tasks, and each task is a pair where the first element represents the machine number that can execute the task and the second is the duration of the task.

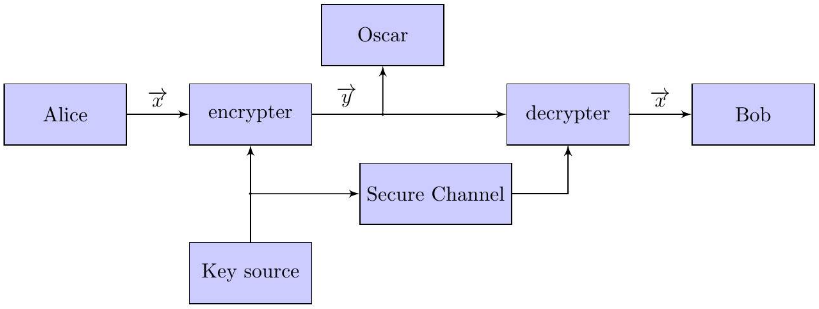
Simple Job Shop Problem Solution in Z3 SMT-LIB

```
;;; https://www.cs.toronto.edu/˜victorn/tutorials/z3/SMT.html
;;; Variables representing the starting times of the task:
(declare-const t00 Int)
(declare-const t01 Int)
(declare-const t02 Int)
(declare-const t10 Int)
(declare-const t11 Int)
(declare-const t12 Int)
(declare-const t20 Int)
(declare-const t21 Int)
```

```
;;; A first constraint we need to add is that all starting times
;;; for the tasks are positive:
(assert (and (>= t00 0) (>= t01 0) (>= t02 0)
             (>= t10 0) (>= t11 0) (>= t12 0)
         (>= t20 0) (>= t21 0)))
;;; "All tasks in a job should be executed sequentially" constraints:
(assert (and (<= (+ t00 3)  t01) (<= (+ t01 2)  t02)))
(assert (and (<= (+ t10 2)  t11) (<= (+ t11 1)  t12)))
(assert (and (<= (+ t20 4)  t21)))
;;; "A machine can execute only one task at a time" constraints:
;; Machine 0- only 2 tasks, it's easy
(assert (or (and  (<= (+ t00 3) t10))
            (and  (<= (+ t10 2) t00))))
;; Machine 1: three tasks, so 6 possible sequences!
(assert (or (and  (<= (+ t01 2) t12) (<= (+ t12 4) t20))
            (and  (<= (+ t01 2) t20) (<= (+ t20 4) t12))
            (and  (<= (+ t12 4) t01) (<= (+ t01 2) t20))
            (and  (<= (+ t12 4) t20) (<= (+ t20 4) t01))
            (and  (<= (+ t20 4) t01) (<= (+ t01 2) t12))
            (and  (<= (+ t20 4) t12) (<= (+ t12 4) t01))))
;; Machine 2 : three tasks, so 6 possible sequences!
(assert (or (and  (<= (+ t02 2) t11) (<= (+ t11 1) t21))
            (and  (<= (+ t02 2) t21) (<= (+ t21 3) t11))
            (and  (<= (+ t11 1) t02) (<= (+ t02 2) t21))
            (and  (<= (+ t11 1) t21) (<= (+ t21 3) t02))
            (and  (<= (+ t21 3) t02) (<= (+ t02 2) t11))
            (and  (<= (+ t21 3) t11) (<= (+ t11 1) t02))))
(check-sat)
(get-model)    ;;; Not an optimal model
;; define the max function (to have less idle time)
(define-fun max ((x Int) (y Int)) Int (ite (< x y) y x))
;; optimisation problem
(minimize (max (+ t02 2) (max (+ t12 4) (+ t21 3))))
(check-sat)
(get-model)
(exit)
```

§3.8 Application: Cryptography

Cryptography is the study of methods for sending secret messages. It involves ***encryption***, in which a message, called ***plaintext***, is converted into a form, called ***ciphertext***, that may be sent over channels possibly open to view by outside parties. The receiver of the ciphertext uses ***decryption*** to convert the ciphertext back into plaintext. Pictorially, we have



Cryptography is important today because computer security is crucial to business activities and computer communication. Modular arithmetic is frequently used to append an extra check digit to identification numbers, in order to recognise transmission errors or forgeries. Personal identification numbers of some kind appear on passports, credit cards, bank accounts, and a variety of other settings.

Definition 3.8.1. A *cryptosystem* is a tuple $(\mathscr{P},\mathscr{C},\mathscr{K},\mathscr{E},\mathscr{D})$ where

1. $\mathscr{P}$ is a finite set of possible plaintexts;
2. $\mathscr{C}$ is a finite set of possible ciphertexts;
3. $\mathscr{K}$ is a set of possible keys called the *keyspace*;
4. For each $k \in \mathscr{K}$, there is an encryption rule and a decryption rule respectively as follows:

   $$e_K : \mathscr{P} \to \mathscr{C}, \quad d_K : \mathscr{C} \to \mathscr{P}$$

   such that $d_K(e_K(x)) = x$ for every $x \in \mathscr{P}$. The set of e_K is denoted $\mathscr{E}$ and the set of d_K is denoted $\mathscr{D}$.

Remark 3.8.2. When $\mathscr{P} = \mathscr{C}$, then each encryption function is in fact a permutation.

In the following subsections, we will investigate three *classical (or private-key, symmetric-key) cryptosystem* and one public-key cryptosystem, the RSA cryptosystem. For simplicity, we limit ourselves to Latin characters A to Z ignoring the difference between capital and small letters. We assume each letter of the alphabet is coded by its position relative to the others as follows:

`A=0, B=1, C=2, D=3,` $\cdots$`, X=23, Y=24, Z=25.`

§3.8.1 Shift Cipher

Definition 3.8.3. Let $\mathscr{P} = \mathscr{C} = \mathscr{K} = \mathbb{Z}_{26}$. For $0 \leq k \leq 25$, define

$$e_k(x) = (x + k) \mod 26, \quad d_k(y) = (y - k) \mod 26$$

for every x, y in $\mathbb{Z}_{26}$. Then $(\mathscr{P},\mathscr{C},\mathscr{K},\mathscr{E},\mathscr{D})$ is called a *shift cipher*.

Example 3.8.4. An encryption system once used by Julius Caesar, and now called the Caesar cipher, encrypts messages by changing each letter of the alphabet to the one three places farther along, with X wrapping around to A, Y to B, and Z to C.

If the numerical version of the plaintext for a letter is denoted M and the numeric version of the ciphertext is denoted C, then

$$C = (M + 3) \mod 26$$

The receiver of such a message can easily decrypt it by using the formula

$$M = (C - 3) \mod 26$$

When a private key cryptosystem is used, a pair of people who wish to communicate in secret must have a separate key. Since anyone knowing this key can both encrypt and decrypt messages easily, these two people need to securely exchange the key.

Example 3.8.5. Use the Caesar cipher to encrypt the message `HOW ARE YOU`.

Solution: First translate the letters of `HOW ARE YOU` into their numeric equivalents:

`7 14 22 0 17 4 24 14 20`

Next encrypt the message by adding 3 to each number.

`10 17 25 3 20 7 1 17 23`

Finally, substitute the letters that correspond to these numbers. The encrypted messages becomes

`KRZ DUH BRX`

Example 3.8.6. Use the shift cipher with key 3 to decrypt the message `L DP ILQH`.

Shift ciphers can be broken by what we call a *brute force attack*. It only takes 25 trials to guess the private key, hence, this is a useless cryptosystem in an information society.

Example 3.8.7. Consider Bob received a cipher text "`haahjr ha khdu`" encrypted using shift cipher. What is the original message?

Solution: Apply every possible decryption key from 1 through 25. Look at the results and see which one makes sense:

```
Shift: 0:  haahjr ha khdu
Shift: 1:  ibbiks ib liev
Shift: 2:  jccjlt jc mjfw
Shift: 3:  kddkmu kd nkgx
Shift: 4:  leelnv le olhy
Shift: 5:  mffmow mf pmiz
Shift: 6:  nggnpx ng qnja
Shift: 7:  ohhoqy oh rokb
Shift: 8:  piiprz pi splc
Shift: 9:  qjjqsa qj tqmd
Shift: 10: rkkrtb rk urne
Shift: 11: sllsuc sl vsof
Shift: 12: tmmtvd tm wtpg
Shift: 13: unnuwe un xuqh
Shift: 14: voovxf vo yvri
Shift: 15: wppwyg wp zwsj
Shift: 16: xqqxzh xq axtk
Shift: 17: yrryai yr byul
Shift: 18: zsszbj zs czvm
Shift: 19: attack at dawn
Shift: 20: buubdl bu ebxo
Shift: 21: cvvcem cv fcyp
Shift: 22: dwwdfn dw gdzq
Shift: 23: exxego ex hear
Shift: 24: fyyfhp fy ifbs
Shift: 25: gzzgiq gz jgct
```

§3.8.2 Affine Cipher

Definition 3.8.8. Let $\mathscr{P} = \mathscr{C} = \mathbb{Z}_{26}$ and $\mathscr{K} = \{(a, b) \in \mathbb{Z}_{26} \times \mathbb{Z}_{26} : \gcd(a, 26) = 1\}$. For each permutation $(a, b) \in \mathscr{K}$, define

$$e_{a,b}(x) = (ax + b) \mod 26, \quad d_{a,b}(y) = a^{-1}(y - b) \mod 26.$$

Since $\gcd(a, 26) = 1$, a can only take values from 1,3,5,7,9,11,15, 17,19,21,23,25.

Example 3.8.9. Encipher "ITS COOL" using an affine cipher with $a = 5$ and $b = 8$.

Solution: Using $e_{5,8}(x) = (5x + 8) \mod 26$, we fill in the following table

plaintext	I	T	S	C	O	O	L
x	8	19	18	2	14	14	11
$5x + 8$	48	103	98	18	78	78	63
$(5x + 8) \mod 26$	22	25	20	18	0	0	11
ciphertext	W	Z	U	S	A	A	L

Example 3.8.10. Decipher "HPCCXAQ" using an affine cipher with $a = 5$ and $b = 8$.

§3.8.3 Substitution Cipher

A substitution cipher is one in which letters are represented by other letters; it can be decriphered by someone knowing the order of the cipher alphabet used. It is defined formally as follows.

Definition 3.8.11. Let $\mathscr{P} = \mathscr{C} = \mathbb{Z}_{26}$ and $\mathscr{K}$ be the set of all possible permutations of the 26 symbols in $\mathscr{P}$. For each substitution $\sigma \in \mathscr{K}$, define

$$e_\sigma(x) = \sigma(x), \quad d_\sigma(y) = \sigma^{-1}(y).$$

Remark 3.8.12. There are 26! permutations. Hence, finding the right private key may be difficult.

Example 3.8.13. Consinder the following permutation for substitution cipher:

$$\begin{pmatrix} A & B & C & D & E & F & G & H & I & J & K & L & M & N & O & P & Q & R & S & T & U & V & W & X & Y & Z \\ R & Z & B & U & Q & K & F & C & P & Y & E & V & L & S & N & G & W & O & X & D & J & I & A & H & T & M \end{pmatrix}$$

Encode the word "HARDWORKING".

Substitution ciphers are fairly easy to "crack" — the problem is that in English (or any language), certain letters are far more likely to appear. In English, for example, the letter "E" is far more likely to appear than the letter "Z". In fact, we have the following English letter frequency table

A	8.2%	F	2.2%	K	0.8%	P	1.9%	U	2.8%	Z	0.1%
B	1.5%	G	2.0%	L	4.0%	Q	0.1%	V	1.0%		
C	2.8%	H	6.1%	M	2.4%	R	6.0%	W	2.3%		
D	4.3%	I	7.0%	N	6.7%	S	6.3%	X	0.1%		
E	12.7%	J	2.2%	O	7.5%	T	9.1%	Y	2.0%		

The approximate percentages for the first few letters in the list below are:

E: 12.7%, T: 9.1%, A: 8.2%, O: 7.5%

and the percentages for the last few are:

J: 0.2%, Q: 0.1%, Z: 0.1%.

§3.8.4 RSA Cryptosystem

RSA is a popular public-key encryption method used in electronic commerce. In what follows, we will learn how to encrypt and decrypt a message using RSA cryptography. First, we define RSA formally.

Definition 3.8.14. Let $n = pq$, where p and q different prime numbers. Let $\mathscr{P} = \mathscr{C} = \mathbb{Z}_n$, define

$$\mathscr{K} = \{(n, p, q, a, b) : ab \equiv 1 \pmod{\phi(n)}\}.$$

For every $k \in \mathscr{K}$, we define

$$e_k(x) = x^b \mod n, \quad d_k(y) = y^a \mod n$$

where $x, y \in \mathbb{Z}_n$ and ϕ is the ***Euler phi function***, which is an arithmetic function that counts the number of positive integers less than or equal to n that are relatively prime to n. It is found mathematically to be

$$\phi(n) = n \prod_{\substack{p|n \\ p \text{ is prime}}} \left(1 - \frac{1}{p}\right).$$

The values n and b comprise the *public key* and the values p, q and a form the *private key.*

Example 3.8.15 (Getting familiar with Euler ϕ function)**.** Find the number of integers relatively prime to 36.

Solution:

$$\phi(36) = \phi(2^2 3^2) = 36\left(1 - \frac{1}{2}\right)\left(1 - \frac{1}{3}\right) = 36 \cdot \frac{1}{2} \cdot \frac{2}{3} = 12.$$

In words, this says that the distinct prime factors of 36 are 2 and 3; half of the thirty-six integers from 1 to 36 are divisible by 2, leaving eighteen; a third of those are divisible by 3, leaving twelve coprime to 36. And indeed there are twelve: 1, 5, 7, 11, 13, 17, 19, 23, 25, 29, 31, and 35.

Theorem 3.8.16. *When p and q are prime numbers, $\phi(pq) = (p-1)(q-1)$.*

To encrypt a message using the RSA cipher, a person needs to know the value of pq and of another number b, both of which are made publicly available. But only a person who knows the individual values of p, q and a can decrypt an encrypted message.

Example 3.8.17. Suppose Alice decides to set up an RSA cipher. She chooses two prime numbers, say $p = 5$ and $q = 11$, and computes $n = pq = 55$. She then chooses a positive integer b that is relatively prime to $(p-1)(q-1)$. In this case, $(p-1)(q-1) = 4(10) = 40$, so she may take $b = 3$ is relatively prime to 40. The two numbers $n = 55$ and $b = 3$ are the public key, which she may distribute widely. To decrypt the message, Alice needs to find the decryption key, a number a that is a positive inverse to b modulo $(p-1)(q-1)$. In this case, the key is

$$k = (55, 5, 11, a, 3).$$

1. Bobs wants to send Alice the message `HA`. Find the ciphertext for his message.
2. Find the value of a and decrypt the ciphertext 17.

Solution: Given $p = 5$, $q = 11$, $n = pq = 55$, $b = 3$, $e_k(x) = x^3 \mod n$, $d_k(y) = y^a \mod n$.

1. Bob will send his message in two blocks, one for the `H` and another for the `A`. The letters `H` and `A` are encoded as `7` and `0` respectively. The corresponding ciphertext is computed as follows:

$$\begin{aligned} e(7) &= 7^3 \mod 55 = 343 \mod 55 = 13, \\ e(0) &= 0^3 \mod 55 = 0. \end{aligned}$$

   Accordingly, Bob sends Alice the message: `13 0`.

2. The integer a needs to satisfy

$$ab = 3a \equiv 1 \pmod{\phi(55)}$$

   Here, $\phi(55) = (p-1)(q-1) = 40$. This problem is similar to Example 3.6.27(2), in which we found that

$$a \equiv 3^{-1} \equiv 27 \pmod{40}.$$

   Then we compute

$$\begin{aligned} d(17) &\equiv 17^{27} \\ &\equiv 17^{16+8+2+1} \\ &\equiv [(17^{16} \mod 55)(17^8 \mod 55)(17^2 \mod 55)(17 \mod 55)] \mod 55 \\ &\equiv (16 \cdot 26 \cdot 14 \cdot 17) \mod 55 \\ &\equiv 99008 \equiv 8 \pmod{55} \end{aligned}$$

$$\text{where} \begin{cases} 17^2 \mod 55 = 17^2 \mod 55 = 14 \\ 17^4 \mod 55 = (14)^2 \mod 55 = 31 \\ 17^8 \mod 55 = (31)^2 \mod 55 = 26 \\ 17^{16} \mod 55 = (26)^2 \mod 55 = 16. \end{cases}$$

   Thus the plaintext of Bob's message is 8. The letter corresponding to 8 is `I`.

In reality, RSA is used in setting up a secure communication channel as described in the following example. These days, a key length of at least 2048 bits is required (`https://stackoverflow.com/questions/589834/what-rsa-key-length-should-i-use-for-my-ssl-certificates`).

The Transport Layer Security, TLS for short (replaces the older Secure Socket Layer (SSL)), is a protocol by which many services that communicate over the Internet can do so in a secure fashion. Before we discuss how TLS works and what kinds of security it provides, let us first see what happens without TLS.

Life on the Internet without TLS

Let us compare communications between computers on the Internet and communications between people over the telephone. Without TLS, your computer-to-computer communications suffer from the same security problems from which your telephone communications suffer:

- **Who are you talking to**? In a phone conversation, how can you be sure that the person who picks up the phone at the other end is really the person you are trying to call (especially if you have never spoken to them before)? What if your phone call was intercepted or re-routed, or what if someone else is answering your call recipient's phone? There really is no way to be sure you have reached the right person, especially if they are actively trying to fool you (as what we seen in movies).

- Eavesdropping? As you are aware of from watching TV or reading, **it is very easy to tap phone lines**: the police and spies do this all the time to covertly gather information. It is not easy to detect if your lines are tapped. The same applies with communications over the Internet — how can you be sure that your communications are not being "tapped" and recorded? This is especially problematic in public wifi hotspots.

This results in two very real security issues for communications over the Internet: 1. knowing for sure that you are connecting to the right servers (i.e. those at your bank and not those at a hacker's or phisher's web site), and 2. knowing that your data is safe from prying eyes during transit to those computers. This is where TLS comes in.

Enter the TLS (`https://en.wikipedia.org/wiki/Transport_Layer_Security`)

To solve these problems to a large degree, most Internet services support use of TLS as a mechanism for securing communications. To illustrate how TLS works, let us use another analogy.

Client wants to communicate with a company to send important information back and forth. Client wants to be 100% sure that s/he is communicating with this particular company and that no one can eavesdrop on or intercept the communications. How can s/he do this?

- Client sends a courier to the company's address.
- The company has envelopes that, when closed, can only be opened by the company. The company and the courier go together to a trusted third party — a notary — which makes the company provide documentation to prove its identity. The notary certifies the company's secure envelopes and the courier takes these back to the client.
- The client gets the envelopes and, if it trusts the notary's reputation, can be sure that they are actually from the company indicated.
- The client also has secure envelopes that, once sealed, only the client can open. It puts some of these in one of the company's secure envelopes and sends them back to the company.
- The company gets the sealed secure envelope. It opens the envelope (as only it can). It now has the client's secure envelopes.
- The company has another kind of envelope that can be opened and sealed only by using a special combination. The company puts this special envelope with the combination lock, together with the combination, into one of the client's secure envelopes. The company seals the envelope.
- The company has another type of secure envelope that anyone can open, but which only the company can seal. If you open one of these sealed envelopes, you know for sure that it was sent by the company. The company puts the whole package inside this and sends it to the client.
- When the client gets the secure envelope, it opens it and thus knows that it came from the company. It then opens the next secure envelope inside that can only be opened by the client. Inside it gets out the combination-envelope and the combination itself.
- The client the puts his data in the combination envelope, seals it and sends it to the company.
- The company receives it, opens it, and puts the response in the same secure envelope and sends it back.
- The procedure is repeated as often as necessary for required communications.

TLS relies on **public key cryptography** (e.g. RSA) to accomplish these tasks. In normal encryption, the two parties communicating share a "password" and that password is used to both encrypt and decrypt messages. While this is fast and efficient, how do you communicate these passwords to people you have not yet met in a way that is itself secure?

In "public key cryptography", each person has two keys — a **public** key and a **private** key. Anything encrypted with the user's public key can only be decrypted with the private key and vice versa. Each person then tells the world what his public key is and keeps his private key safe and secure, and private.

If John sends Mary a message encrypted with Mary's public key, then only Mary can open it, as only she has her private key. This is like an envelope that anyone can seal but which only Mary can open.

If John sends Mary a message encrypted with John's private key, then anyone can open it, as everyone has access to John's public key. However, successfully opening the message proves that it was sent by John and no one else, as only John has access to his private key. This is like an envelope that only John can seal, but which anyone can open and thus prove that John sealed it.

TLS in Action

So, let's see how TLS actually works for securing your communications over the Internet. Before the communications occur, the following takes place:

- A company wishes to secure communications to their server `company.com`.
- They create a public and private key for `company.com` (this is also known as as **SSL Certificate**):

```
openssl req -out CSR.csr -new -newkey rsa:2048 -nodes -keyout privateKey.key
```

- They go to a trusted third party company such as Thawte or Verisign: Thawte makes the company prove its identity and right to use the company.com domain. This usually involves a lot of paperwork and paying a hefty fee.
- Once the verification is complete, Thawte gives the company a new public key that has some additional information in it. This information is the certification from Thawte that this public key is for the company and `company.com` and that this is verified by Thawte. This certification information is encrypted using Thawte's private key.

Then, when Client wishes to communicate with the company at `company.com`,

- Client makes a connection to `company.com` with its computer. This connection is made to a special "port" (address) on `company.com` that is set up for TLS communications only.
- When Client connects to `company.com` on its TLS-secured port, the company sends back its public key (and some other information, like what Ciphers it supports).
- Client gets the public key and decides if it is OK
  - If the public key has expired, this could be a problem
  - If the public key claims to be for some domain that is not `company.com` that could be a problem.
  - Client has the public key for Thawte (and many other third party companies) stored in its computer — because these come with the computer. Thus, client can decrypt the validation information, prove the validation is from Thawte and verify that the public key is certified by Thawte. If Client trusts Thawte, then Client can trust that he/she is really communicating with Company. If Client doesn't trust Thawte, or whatever Third Party company is actually being used, then the identity of who is running the computers to which Client is connecting is suspect.
- If the client doesn't trust the server, then the communication is terminated.

- If the client has its own SSL certificate installed, it may send that to the server at this point to see if the server trusts the client. Client-side SSL certificates are not commonly used, but provide a good way for the client to authenticate itself with the server without using a username or password. In the case where this is used, the server would have to know about the client's certificate and verify it in a similar way to how the client verified the server. If this fails, the connection is terminated. If a client-side certificate is not needed, this step is skipped.
- Once the client is happy with the server (and the server with the client, if needed), then the client choose an TLS Cipher to use from the list of encryption methods provided by the server, and generates a "symmetric key" (password) for use with that Cipher. The client encrypts this password using the server's public key and sends it back to the server. The server (and only the server) can decrypt this message and get this password, which is now shared by both the client and server.
- The client will then start communicating with the company by encrypting all data using this password and the chosen Cipher. Normal "symmetric" (password-based) encryption takes place from this point forward because it is much faster than using the public and private keys for everything. These keys were needed to enable the company (and possibly the client) to prove its identity and right to domain.com and to enable the client and server to generate and securely communicate a common password.

§3.9 Application: Random Number Generator and Simulation

`https://en.wikipedia.org/wiki/Random_number_generation` is a process by which, often by means of a random number generator (RNG), a sequence of numbers is generated that cannot be reasonably predicted better than by random chance.

True random number generators can be **hardware random-number generators (HRNGs)**, wherein each generation is a function of the current value of a physical environment's attribute that is constantly changing in a manner that is practically impossible to model.

`https://en.wikipedia.org/wiki/Pseudorandom_number_generator` is a **computer algorithm for generating a sequence of numbers** whose properties **approximate** the properties of sequences of **random numbers**. It is important in practice for their speed in number generation and their reproducibility.

Random number generators have applications in statistical sampling, computer simulation, cryptography, completely randomised design (for agriculture/scientific experiments), and other areas where producing an unpredictable result is desirable. Generally, in applications having unpredictability as the paramount feature, such as in security applications, hardware generators are generally preferred over pseudorandom algorithms, where feasible.

`http://en.wikipedia.org/wiki/Linear_congruential_generator` (LCG) is the most popular random number generator, which is one of the oldest algorithm based on modular arithmetic (Section 3.6) and **recurrence relation**:

$$X_{n+1} = (aX_n + c) \mod m \tag{3.9}$$

where X_n is the sequence of pseudo-random values, and $m > 0$ is the **modulus**, $0 < a < m$ is the **multiplier**, $0 \leq c < m$ is the **increment**, $0 \leq X_0 < m$ is the **seed** respectively. They are integer constants that specify the generator. If $c = 0$, the generator is often called a ***multiplicative congruential method***, or ***Lehmer RNG***. If $c \neq 0$, the generator is called a ***mixed congruential method***.

Only m different values are possible, the period surely cannot be longer than m. Can we achieve the maximum length, m? The following theorem provides a way for us to check if the maximum period is achieved.

Theorem 3.9.1. *The linear congruential sequence defined by* (3.9) *has a period length m if and only if*

1. *c is relatively prime to m;*

2. *$b = a - 1$ is a multiple of p, for every prime p dividing m;*

3. *b is a multiple of 4, if m is a multiple of 4.*

Example 3.9.2. Consider $a = 5$, $c = 1$, and $m = 8$. If the seed X_0 is set to 2, find the resulting sequence.

Efficient LCGs have an m equal to a power of 2, most often $m = 2^{32}$ or $m = 2^{64}$, because this allows the modulus operation to be computed by merely truncating all but the rightmost 32 or 64 bits. The following table lists the parameters of LCGs in common use, including built-in `rand()` functions in runtime libraries of various compilers.

LCGs are fast and require minimal memory (typically 32 or 64 bits) to retain state. This makes them valuable for simulating multiple independent streams.

However, LCGs should not be used for applications where high-quality randomness is critical. For example, it is not suitable for a Monte Carlo simulation because of the serial correlation (among other things). They ***should not be used for cryptographic applications***; see `http://en.wikipedia.org/wiki/Cryptographically_secure_pseudo-random_number_generator` for more suitable generators. If a LCG is seeded with a character and then iterated once, the result is a simple classical cipher called an affine cipher; this cipher is easily broken by standard frequency analysis.

A further problem of LCGs is that the lower-order bits of the generated sequence have a far shorter period than the sequence as a whole if m is set to a power of 2. In general, the nth least significant digit in the base b representation of the output sequence, where $b^k = m$ for some integer k, repeats with at most period b^n.

Nevertheless, LCGs may be a good option. For instance, in an embedded system, the amount of memory available is often severely limited. Similarly, in an environment such as a video game console taking a small number of high-order bits of an LCG may well suffice. The low-order bits of LCGs when m is a power of 2 should never be relied on for any degree of randomness whatsoever. Indeed, simply substituting 2^n for the modulus term reveals that the low order bits go through very short cycles. In particular, any full-cycle LCG when m is a power of 2 will produce alternately odd and even results.

The following are random number generators supported by C++11 `<random>`:

- `linear_congruential_engine`: implements linear congruential algorithm

- `mersenne_twister_engine`: implements Mersenne twister algorithm (used in Python)

- `subtract_with_carry_engine`: implements a subtract-with-carry (lagged Fibonacci) algorithm

- `philox_engine` (C++26): a counter-based parallelizable generator

The following are predefined random number generators by C++:

- `default_random_engine` (C++11): It defaults to using `minstd_rand0` or CPU random generator (e.g. Intel `https://en.wikipedia.org/wiki/RDRAND` on GNU/Linux is shown as `/dev/random`).
- `minstd_rand0` (C++11): discovered in 1969 by Lewis, Goodman and Miller & adopted as "Minimal standard" in 1988 by Park and Miller `std::linear_congruential_engine<std::uint_fast32_t, 16807, 0, 2147483647>`
- `minstd_rand` (C++11): a newer "Minimum standard", recommended by Park, Miller, and Stockmeyer in 1993 `std::linear_congruential_engine<std::uint_fast32_t, 48271, 0, 2147483647>`

- `mt19937` (C++11): a 32-bit Mersenne Twister by Matsumoto and Nishimura in 1998 `std::mersenne_twister_engine<std::uint_fast32_t, 32, 624, 397, 31, 0x9908b0df, 11, 0xffffffff, 7, 0x9d2c5680, 15, 0xefc60000, 18, 1812433253>`
- `mt19937_64` (C++11): a 64-bit Mersenne Twister by Matsumoto and Nishimura, 2000 `std::mersenne_twister_engine<std::uint_fast64_t, 64, 312, 156, 31, 0xb5026f5aa96619e9, 29, 0x5555555555555555, 17, 0x71d67fffeda60000, 37, 0xfff7eee000000000, 43, 6364136223846793005>`
- `ranlux24_base` (C++11): `std::subtract_with_carry_engine<std::uint_fast32_t, 24, 10, 24>`
- `ranlux48_base` (C++11): `std::subtract_with_carry_engine<std::uint_fast64_t, 48, 5, 12>`
- `ranlux24` (C++11): a 24-bit RANLUX generator by Martin Lüscher and Fred James, 1994 `std::discard_block_engine<std::ranlux24_base, 223, 23>`
- `ranlux48` (C++11): a 48-bit RANLUX generator by Martin Lüscher and Fred James, 1994 `std::discard_block_engine<std::ranlux48_base, 389, 11>`
- `knuth_b` (C++11): `std::shuffle_order_engine<std::minstd_rand0, 256>`
- `philox4x32` (C++26): `std::philox_engine<std::uint_fast32_t, 32, 4, 10, 0xCD9E8D57, 0x9E3779B9, 0xD2511F53, 0xBB67AE85>`
- `philox4x64` (C++26): `std::philox_engine<std::uint_fast64_t, 64, 4, 10, 0xCA5A826395121157, 0x9E3779B97F4A7C15, 0xD2E7470EE14C6C93, 0xBB67AE8584CAA73B>`

In statistics, a *sample* is a subject chosen from a *population for investigation*; a *random sample* is one chosen by a method involving an unpredictable component. Random sampling can also refer to taking a number of independent observations from the same probability distribution, without involving any real population. The sample usually is not a representative of the population of people from which it was drawn — this random variation in the results is termed as *sampling error*. One method of producing random samples is by using LCG.

Example 3.9.3. Consider a small town of 25 people with the following salaries:

3000	2100	5100	3400	6000	2500	4800	3900	5100	3200
4900	5600	4300	5800	3400	3300	5300	5400	2100	2400
2300	4000	4100	3400	3600					

A company only has the budget to survey 5 person, use the LCG with $m = 25$, $c = 3$, $a = 6$ and the seed $X_0 = 2$ to find the five random values, the sample average and compare it to the population average. [Note: The first person is labelled with number 0, the second is labelled number 1, etc.]

Example 3.9.4. A C++ program to generate 10000 normally distributed random numbers with a mean of 10 and a standard deviation of 3.

Solution

```
// https://en.cppreference.com/w/cpp/numeric/random.html
#include <random>
#include <map>      // for histogram
#include <iomanip>
```

```
#include <iostream>
#include <string>

int main()
{
    // std::seed_seq consumes a sequence of integer-valued data and
    // produces a requested number of 32-bit unsigned integer values,
    // based on the consumed data.
    std::seed_seq seedno{2025, 2035, 2015, 2075};
    std::mt19937 rand_engine(seedno);
    // Generate a normal distribution
    const double mean = 10, stdev = 3;
    std::normal_distribution<> rnorm(mean, stdev);

    std::map<int, int> hist;
    for (int n = 0; n < 10000; ++n)
        ++hist[std::round(rnorm(rand_engine))];

    std::cout << "Normal(" << mean << ", " << stdev << "):\n"
              << std::fixed << std::setprecision(1);
    for (auto [x, y] : hist)
        std::cout << std::setw(2) << x << ' ' << std::string(y / 100, '*') << '\n';
}
```

Output of the C++ program

```
Normal(10, 3):
-1
 0
 1
 2
 3
 4 **
 5 ***
 6 *****
 7 ********
 8 ***********
 9 *************
10 *************
11 ************
12 **********
13 *******
14 *****
15 ***
16 *
17
18
19
20
```

Exercise with Past Year Questions

Example 3.9.5 (Final May 2021 Sem, Q3 (during MCO)). (a) Use direct proof to show that if n is an odd number, then $n^3 + n$ is even. (3 marks)

Solution: By definition, if n is odd, there is a k such that $n = 2k + 1$. . . . . [1 mark]

$$\begin{aligned} n^3 + n &= (2k+1)^3 + (2k+1) \\ &= 8k^3 + 3 \times 4k^2 + 3 \times 2k + 1 + 2k + 1 \qquad [1.5 \text{ marks}] \\ &= 2(4k^3 + 6k^2 + 4k + 1) \end{aligned}$$

Since $4k^3 + 6k^2 + 4k + 1$ is an integer, by definition, $n^3 + n$ is even. . . . . [0.5 mark]

(b) Prove that $\sqrt{6}$ is irrational using the method of contradiction. Hence, use the result to prove that $\sqrt{2} + \sqrt{3}$ is irrational. (3 marks)

Solution: Suppose that $\sqrt{6}$ is rational and it can be written as a ratio of two relatively prime integers p and q:

$$\sqrt{6} = \frac{p}{q} \qquad [0.5 \text{ mark}]$$

By squaring both sides and rearranging terms, we have

$$6q^2 = p^2. \qquad [0.5 \text{ mark}]$$

The left-hand-side indicate that p^2 is an even number, hence, p is even (otherwise, we will have a contradiction): There is an integer k such that

$$p = 2k \qquad [0.5 \text{ mark}]$$

Therefore

$$6q^2 = 4k^2 \Rightarrow 3q^2 = 2k^2. \qquad [0.5 \text{ mark}]$$

Since $3q^2$ is even, q has to be even (otherwise a contradiction). There is an integer k_2 such that

$$q = 2k_2. \qquad [0.5 \text{ mark}]$$

This is a contradiction to the fact that $\sqrt{6}$ is rational with p and q relatively prime.

Suppose $\sqrt{2} + \sqrt{3}$ is rational and can be expressed as

$$\sqrt{2} + \sqrt{3} = \frac{p_2}{q_2}$$

where p_2 and q_2 are some integers. By squaring both sides of the equality, we have

$$2 + 2\sqrt{6} + 3 = \frac{p_2^2}{q_2^2} \Rightarrow \sqrt{6} = \frac{p_2^2 - 5q_2^2}{2q_2^2} \qquad [0.5 \text{ mark}]$$

which contradicts with the fact that $\sqrt{6}$ is irrational.

(c) Use mathematical induction to prove that for all integers $n \geq 1$,

$$1 + 4 + 7 + \cdots + (3n-2) = \frac{n(3n-1)}{2}. \qquad (4 \text{ marks})$$

Solution: Let the predicate $P(n)$ be $1 + 4 + 7 + \cdots + (3n - 2) = \frac{n(3n-1)}{2}$.

Base step: When $n = 1$,

$$RHS = \frac{1(3-1)}{2} = 1 = LHS \qquad \text{[1 mark]}$$

Inductive step: Suppose that the predicate $P(k)$ is valid when $n = k$, i.e.

$$1 + 4 + 7 + \cdots + (3k - 2) = \frac{k(3k-1)}{2}. \qquad \text{[0.5 mark]}$$

We want to show that the predicate $P(k)$ implies $P(k+1)$:

$$\begin{aligned}
\text{LHS of } P(k+1) &= 1 + 4 + 7 + \cdots + (3k-2) + (3(k+1) - 2) && \text{[0.5 mark]}\\
&= \underbrace{\frac{k(3k-1)}{2}}_{\text{using } P(k)} + 3k + 1 && \text{[0.5 mark]}\\
&= \frac{k(3k-1) + 6k + 2}{2} && \text{[0.5 mark]}\\
&= \frac{3k^2 + 5k + 2}{2} = \frac{(k+1)(3k+2)}{2} && \text{[0.5 mark]}\\
&= \frac{(k+1)(3(k+1) - 1)}{2} = \text{RHS of } P(k+1) && \text{[0.5 mark]}
\end{aligned}$$

Therefore $P(k)$ implies $P(k+1)$. By the principle of mathematical induction, for all $n \geq 1$, $P(n)$ is true.

[Total: 10 marks]

Exercise with Past Year Questions

Only 2021 questions are set by me. The rest are by other lecturers.

UECM1304 Jan 2021 Semester

Example 3.9.6 (Final May 2021 Sem, Q3 (during MCO)). 1. Use direct proof to show that if n is an odd number, then $n^3 + n$ is even. (3 marks)

Proof: By definition, if n is odd, there is a k such that $n = 2k + 1$. . . . .[1 mark]

$$\begin{aligned}
n^3 + n &= (2k+1)^3 + (2k+1)\\
&= 8k^3 + 3 \times 4k^2 + 3 \times 2k + 1 + 2k + 1 && \text{[1.5 marks]}\\
&= 2(4k^3 + 6k^2 + 4k + 1)
\end{aligned}$$

Since $4k^3 + 6k^2 + 4k + 1$ is an integer, by definition, $n^3 + n$ is even. . . . .[0.5 mark]

2. Prove that $\sqrt{6}$ is irrational using the method of contradiction. Hence, use the result to prove that $\sqrt{2} + \sqrt{3}$ is irrational. (3 marks)

Proof: Suppose that $\sqrt{6}$ is rational and it can be written as a ratio of two relatively prime integers p and q:

$$\sqrt{6} = \frac{p}{q} \qquad \text{[0.5 mark]}$$

By squaring both sides and rearranging terms, we have

$$6q^2 = p^2. \qquad \text{[0.5 mark]}$$

The left-hand-side indicate that p^2 is an even number, hence, p is even (otherwise, we will have a contradiction): There is an integer k such that

$$p = 2k \qquad \text{[0.5 mark]}$$

Therefore

$$6q^2 = 4k^2 \Rightarrow 3q^2 = 2k^2. \qquad \text{[0.5 mark]}$$

Since $3q^2$ is even, q has to be even (otherwise a contradiction). There is an integer k_2 such that

$$q = 2k_2. \qquad \text{[0.5 mark]}$$

This is a contradiction to the fact that $\sqrt{6}$ is rational with p and q relatively prime.

Suppose $\sqrt{2} + \sqrt{3}$ is rational and can be expressed as

$$\sqrt{2} + \sqrt{3} = \frac{p_2}{q_2}$$

where p_2 and q_2 are some integers. By squaring both sides of the equality, we have

$$2 + 2\sqrt{6} + 3 = \frac{p_2^2}{q_2^2} \Rightarrow \sqrt{6} = \frac{p_2^2 - 5q_2^2}{2q_2^2} \qquad \text{[0.5 mark]}$$

which contradicts with the fact that $\sqrt{6}$ is irrational.

3. Use mathematical induction to prove that for all integers $n \geq 1$,

$$1 + 4 + 7 + \cdots + (3n - 2) = \frac{n(3n-1)}{2}. \qquad \text{(4 marks)}$$

Proof: Let the predicate $P(n)$ be $1 + 4 + 7 + \cdots + (3n - 2) = \frac{n(3n-1)}{2}$.

Base step: When $n = 1$,

$$RHS = \frac{1(3-1)}{2} = 1 = LHS \qquad \text{[1 mark]}$$

Inductive step: Suppose that the predicate $P(k)$ is valid when $n = k$, i.e.

$$1 + 4 + 7 + \cdots + (3k - 2) = \frac{k(3k-1)}{2}. \qquad \text{[0.5 mark]}$$

We want to show that the predicate $P(k)$ implies $P(k + 1)$:

LHS of $P(k + 1) = 1 + 4 + 7 + \cdots + (3k - 2) + (3(k + 1) - 2)$ [0.5 mark]

$$= \underbrace{\frac{k(3k-1)}{2}}_{\text{using } P(k)} + 3k + 1 \qquad \text{[0.5 mark]}$$

$$= \frac{k(3k-1)+6k+2}{2} \qquad \text{[0.5 mark]}$$

$$= \frac{3k^2+5k+2}{2} = \frac{(k+1)(3k+2)}{2} \qquad \text{[0.5 mark]}$$

$$= \frac{(k+1)(3(k+1)-1)}{2} = \text{RHS of } P(k+1) \qquad \text{[0.5 mark]}$$

Therefore $P(k)$ implies $P(k+1)$. By the principle of mathematical induction, for all $n \geq 1$, $P(n)$ is true.

[Total: 10 marks]

UCCM1363 Jan 2024 Semester

Example 3.9.7 (Final Jan 2024 Sem, Q2)**.** (a) Given the sequence, $a_n = 2a_{n-1}+3$ for all integers $n \geq 1$, $a_1 = 2$.

(i) List FOUR (4) iterations of the sequence by starting from the given initial condition. (4 marks)

Solution: $a_1 = 2$
$a_2 = 2a_1 + 3 = 2(2) + 3 = 7$
$a_3 = 2a_2 + 3 = 2(7) + 3 = 17$
$a_4 = 2a_4 + 3 = 2(17) + 3 = 37$
Standard ML:

```
fun a n = if n<=1 then 2 else 2 * (a (n-1)) + 3;
tl (List.tabulate (5, a));   -- skip a(0)
```

(ii) Use the iterations in (i) to guess an explicit formula of the sequence. (4 marks)

Solution: Suppose $a_n = 2^n \cdot a + b$.
$a_1 = 2^1 \cdot a + b = 2$
$a_2 = 2^2 \cdot a + b = 7$
Solving the above equations gives $(4-2)a = 7-2 = 5 \Rightarrow a = \frac{5}{2}$ and $b = 2 - 2^1 \cdot \frac{5}{2} = -3$.
Therefore $a_n = 5 \cdot 2^{n-1} - 3$.

(b) Propose a formula for the sum of the first n positive odd integers. Then prove your formula using mathematical induction. (9 marks)

Remark: This is basically Example 3.4.2. However, the lecturer setting the question did not write down the formula but ask the students to write it down.

Let $n \geq 1$. $1 + 3 + \cdots + (2n-1) = n^2$ (*)
Proof:
Base step: When $n = 1$, $1 = 1^2$.
Inductive Step: Suppose $1 + 3 + \cdots + (2k-1) = k^2$, we use it on the left-hand-side (LHS) of

$P(k+1)$:

$$\underbrace{1+3+\cdots+(2k-1)}_{\text{LHS of } P(k)}+(2(k+1)-1) = k^2+(2(k+1)-1) = k^2+2k+1 = (k+1)^2 = \text{RHS of } P(k+1).$$

Therefore $P(k) \to P(k+1)$. We have proven that (*) is true for all $n \geq 1$.

(c) Show the following equation using the method of differences.

$$\sum_{r=1}^{n} \frac{2}{r(r+2)} = \frac{3n^2+5n}{2(n+1)(n+2)} \qquad \text{(8 marks)}$$

Proof:

$$\sum_{r=1}^{n} \frac{2}{r(r+2)} = \sum_{r=1}^{n} \left[\frac{1}{r} - \frac{1}{r+2}\right] = \sum_{r=1}^{n} \frac{1}{r} - \sum_{r=1}^{n} \frac{1}{r+2} = \sum_{r=1}^{n} \frac{1}{r} - \sum_{r=3}^{n+2} \frac{1}{r}$$

$$= 1 + \frac{1}{2} - \frac{1}{n+1} - \frac{1}{n+2} = \frac{\frac{3}{2}(n+1)(n+2) - (n+2) - (n+1)}{(n+1)(n+2)}$$

$$= \frac{3(n^2+3n+2) - 2(2n+3)}{2(n+1)(n+2)} = \frac{3n^2+9n+6-4n-6}{2(n+1)(n+2)} = \frac{3n^2+5n}{2(n+1)(n+2)}$$

Exercise: Try to prove it using mathematical induction.

References

S. S. Epp. *Discrete Mathematics with Applications.* Cengage Learning Inc., 5th edition, 2020.

P. Ording. *99 Variations on a Proof.* Princeton University Press, 2019.