Contents

1 Review of Propositional Logic

Task: Recall enough propositional logic to see how it matches up with set theory.

Definition: A proposition is any declarative sentence that is either true or false.

1.1 Connectives

	Connectives		Notation in Maths
and	$\wedge$		
or	$\vee$	"Inclusive or"	
not	$\neg$	Sometimes denoted $\sim$	
implies	$\rightarrow$	if/then; called implication	$\Rightarrow$
if and only if	$\leftrightarrow$	Called equivalence	$\Leftrightarrow$

1.1.1 Truth Table of the Connectives

Let P, Q be propositions:

P	Q	P∧Q
F	F	F
F	T	F
T	F	F
T	T	T

P	Q	P∨Q
F	F	F
F	T	T
T	F	T
T	T	T

P	¬P
F	T
T	F

P	Q	P→Q
F	F	T
F	T	T
T	F	F
T	T	T

P	Q	P↔Q
F	F	T
F	T	F
T	F	F
T	T	T

Priority of the Connectives

Highest to Lowest: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$

1.2 Important Tautologies

$$\left.\begin{array}{lcl} (P \rightarrow Q) & \leftrightarrow & (\neg P \vee Q) \\ (P \leftrightarrow Q) & \leftrightarrow & [(P \rightarrow Q) \wedge (Q \rightarrow P)] \\ \neg(P \wedge Q) & \leftrightarrow & (\neg P \vee \neg Q) \\ \neg(P \vee Q) & \leftrightarrow & (\neg P \wedge \neg Q) \end{array}\right\} \text{ De Morgan Laws}$$

As a result, $\neg$ and $\vee$ together can be used to represent all of $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$.

Less obvious: One connective called the sheffer stroke P|Q (which stands for "not both P and Q" or "P nand Q") can be used to represent all of $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$ since $\neg$P $\leftrightarrow$ P|P and P $\vee$ Q $\leftrightarrow$ (P|P) | (Q|Q).

Recall if P$\rightarrow$Q is a given implication, Q$\rightarrow$P is called the converse or P$\rightarrow$Q. $\neg$Q $\rightarrow$ $\neg$P.

1.3 Indirect Arguments/Proofs by Contradiction/Reductio as absurdum

Based on the tautology $(P \rightarrow Q) \leftrightarrow (\neg Q \rightarrow \neg P)$

Example: Famous argument that $\sqrt{2}$ is irrational.

Proof:

Suppose $\sqrt{2}$ is rational, then it can be expressed is fraction form $\frac{a}{b}$. Let us **assume** that our fraction is in the lowest term, **i.e.** their only common divisor is 1.

Then,

$$\sqrt{2} = \frac{a}{b}$$

Squaring both sides, we have

$$2 = \frac{a^2}{b^2}$$

Multiplying both sides by B^2 yields

$$2b^2 = a^2$$

Since $a^2 b^2$, we can conclude that a^2 is even because whatever the value of b^2 has to be multiplied by 2. If a^2 is even, then a is also even. Since a is even, no matter what the value of a is, we can always find an integer that if we divide a by 2, it is equal to that integer. If we let that integer be k, then $\frac{a}{b} = k$ which means that $a = 2k$.

Substituting the value of $2k$ to a, we have $2b^2 = (2k)^2$ which means that $2b^2 = 4k^2$. dividing both sides by 2 we have $b^2 = 2k^2$. That means that the value b^2 is even, since whatever the value of k you have to multiply it by 2. Again, is b^2 is even, then b is even.

This implies that both a and b are even, which means that both the numerator and the denominator of our fraction are divisible by 2. This contradicts our **assumption** that $\frac{a}{b}$ has no common divisor except 1. Since we found a contradiction, our assumption is, therefore, false. Hence the theorem is true.

qed

2 Predicate logic and Quantifiers

Task: Understand enough predicate logic to make sense of quantified statements.

In predicate logic, propositions depend on variable x, y, z, so their truth value may change depending on which values these variables assume: $P(x), Q(x, y), R(x, y, z)$

2.1 Introduce quantifiers

2.1.1 $\exists$ existential quantifier

Syntax: $\exists x P(x)$

Definition: $\exists x P(x)$ is true if $P(x)$ is true or some value of x; it is false otherwise.

2.1.2 $\forall$ universal quantifier

Syntax: $\forall x P(x)$

Definition: $\forall x P(x)$ is true if $P(x)$ is true for all allowable values of x. It is false otherwise.

2.1.3 $\exists!$ for one and only one

Syntax: $\exists! x P(x)$

Definition: $\exists! x P(x)$ is true if $P(x)$ is true for exactly one value of x and false for all often values of x; otherwise, $\exists! x P(x)$ is false.

2.2 Alternation of Quantifiers

$\forall x \exists y \forall z \quad P(x, y, z)$

NB: The order cannot be exchanged as it might modify the truth values of the statement (think of examples with two quantifiers).

2.3 Negation of Quantifiers

$$\neg(\exists x P(x)) \quad \leftrightarrow \quad \forall x \neg P(x)$$
$$\neg(\forall x P(x)) \quad \leftrightarrow \quad \exists x \neg P(x)$$

3 Set Theory

Task: Understand enough set theory to make sense of other mathematical objects in abstract algebra, graph theory, etc. Set theory started around 1870's $\rightarrow$ late development in mathematics but now taught early in one's maths education due to Bourbaki school.

Definition: A set is a collection of objects. $x \in A$ means the element X is in the set A (**i.e.** belongs to A).

Examples:

1. All students in a class.
2. $\mathbb{N}$ the set of natural numbers starting at 0.

   $\mathbb{N}$ is defined via the following two axioms:

(a) $0 \in \mathbb{N}$
(b) if $x \in \mathbb{N}$ then $x + 1 \in \mathbb{N}$ ($x \in \mathbb{N} \to X + A \in \mathbb{N}$)

3. $\mathbb{R}$ set of real numbers also introduced axiomatically

$\mathbb{R}$ the set of real numbers.
(a) Additive closure: $\forall x, y \exists z(x + y = z)$
(b) Multiplicative closure: $\forall x, y, \exists z(x \times y = z)$
(c) Additive associativity: $x + (y + z) = (x + y) + z$
(d) Multiplicative associativity: $x \times (y \times z) = (x \times y) \times z$
(e) Additive commutativity: $x + y = y + x$
(f) Multiplicative commutativity: $x \times y = y \times x$
(g) Distributivity: $x \times (y + z) = (x \times y) + (x \times z)$ and $(y + z) \times x = (y \times x) + (z \times x)$
(h) Additive identity: There is a number, denoted 0, such that or all $x, x + 0 = x$
(i) Multiplicative identity: There is a number, denoted 1, such that for all $x, x \times 1 = 1 \times x = x$
(j) Additive inverses: For every x there is a number, denoted $-x$, such that $x + (-x) = 0$
(k) Multiplicative inverses: For every nonzero x there is a number, denoted x, such that $x \times x^{-1} = x^{-1} \times x = 1$
(l) $0 \neq 1$
(m) Irreflexivity of $<:\sim (x < x)$
(n) Transitivity of ¡: If $x < y$ and $y < z$, then $x < z$
(o) Trichotomy: Either $x < y, y < x$, or $x = y$
(p) If $x < y$, then $x + y < y + z$
(q) If $x < y$ and $0 < z$, then $x \times z < y \times z$ and $z \times x < z \times y$
(r) Completeness: If a nonempty set of real numbers has an upper bound, then it has a *least* upper bound.

4. $\emptyset$ is the empty set (The set with no elements).

Definition: Let A, B be sets. A=b if and only if all elements of A are elements of B and all elements of B are elements of A,
i.e. $A = b \leftrightarrow [\forall x(x \in A \to x \in B)] \cap [\forall y(y \in B \to y \in A)]$

3.1 Two Ways to Describe Sets

1. The enumeration/roster method: list all elements of the set.
NB: order is irrelevant.
$A = \{0, 1, 2, 3, 4, 5\} = \{5, 0, 2, 3, 1, 4\}$

2. The formulaic/set builder method: give a formula that generates all elements of the set.
$A = \{x \in \mathbb{N} \mid 0 \leq x \wedge x \leq 5\} = \{0, 1, 2, 3, 4, 5\} = \{x \in \mathbb{N} : 0 \leq x \wedge x \leq 5\}$

Using $\mathbb{N}$ and the set-builder method, we can define:

$\mathbb{Z} = \{m - n \mid \forall m, n \in \mathbb{N}\}$

$n = 0$ in any natural numbers $\Rightarrow$ we generate all of $\mathbb{N}$

$m = 0$ in any natural number $\Rightarrow$ we generator all negative integers

$\mathbb{Q} = \{\frac{p}{q} \mid p, q \in \mathbb{Z} \wedge q \neq 0\}$

Definition: A set A is called finite if it has a finite number of elements; otherwise it is called infinite.

4 Set Operations

Task: Understand how to represent sets by Venn diagrams. Understand set union, intersection, complement and difference.

Definition: Let A, B be sets. A is a subset of B. If all elements of A are elements of B, **i.e.** $\forall x(x \in A \rightarrow x \in B)$. We denote that A is a subset of B by $A \subseteq B$

Example: $\mathbb{N} \subseteq \mathbb{Z}$

Definition: Let A, B be sets. A is a proper subset of B if $A \subseteq B \wedge A \neq B$, **i.e.** $A \subseteq B \wedge \exists x \in B s.t. x \notin A$.
A proper subset is always a subset, but a subset is not always a proper subset.

Notation: $A \subset B$

Example: $\mathbb{N} \subset \mathbb{Z}$ since $\exists -1 \in \mathbb{N}$
NB: $\forall A$ a set $\emptyset \subseteq A$

Recall: $B \subseteq C$ means $\forall x(x \in B \rightarrow x \in C)$, but $\emptyset$ has no elements so in $\emptyset \subseteq A$ the quantifier $\forall$ operates on a domain with no elements. Clearly, we need to give meaning to $\exists$ and $\forall$ on empty sets.

Boolean Convention
$\forall$ is true on the empty set
$\exists$ is false on the empty set
} Consistent with common sense

Definition: Let A, B be two sets. The union $A \cup B = \{x \mid x \in A \vee x \in B\}$

Definition: Let A, B be two sets. The intersection $A \cap B = \{x \mid x \in A \wedge x \in B\}$

Definition: Let A, B be sets. A and B are called disjoint is $A \cap B = \emptyset$

Definition Let A, B be two sets. $A - B = A \backslash B = \{a \mid x \in A \wedge x \notin B\}$

Examples:

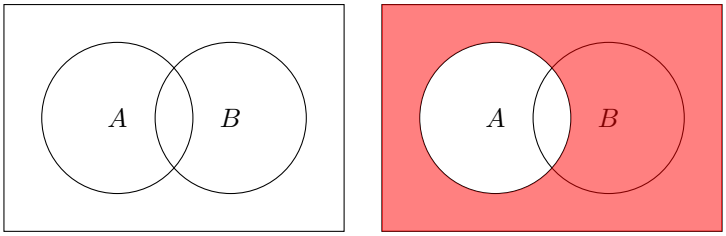
$A = \{1, 2, 5\}$	$B = \{1, 3, 6\}$
$A \cup B = \{1, 2, 3, 5, 6\}$	$A \cap B = \{1\}$
$A \backslash B = \{2, 5\}$	$B \backslash A = \{3, 6\}$

Definition: Let A, U be sets s.t. $A \subseteq U$. The complement of A in $U = U \backslash A = A^C = \{x \mid x \in U \wedge x \notin A\}$

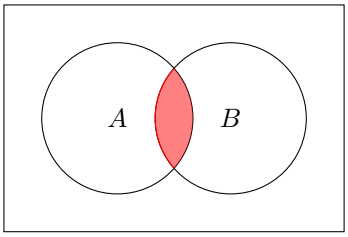
Remark: The notation A^C is unambiguous only if the universe U is clearly defined or understood.

4.1 Venn Diagrams

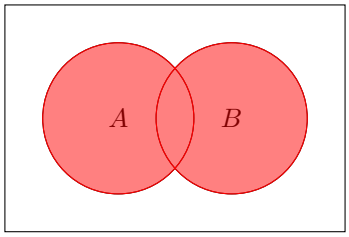
Schematic representation of set operations.



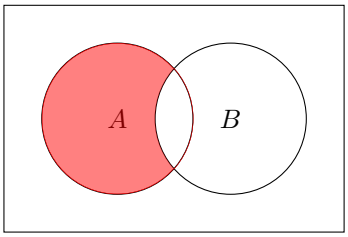
A^C



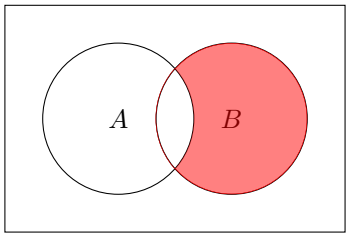
$A \cap B$



$A \cup B$



$A \backslash B$



$B \backslash A$

4.2 Properties of Set Operations

Correspondence between Logic and Set Theory

Logical Connective	Set operation
$\wedge$	intersection $\cap$
$\vee$	union $\cup$
$\neg$	complement $(\ )^C$

As a result, various properties of set operations become obvious:

- Commutativity
  - $A \cap B = B \cap A$
  - $A \cup B = B \cup A$
- Associativity
  - $(A \cup B) \cup C = A \cup (B \cup C)$
  - $(A \cap B) \cap C = A \cap (B \cap C)$
- Distributivity
  - $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
  - $A \cup (B \cap C) = (A \cup B) \cap (A \cup B)$
- De Morgan Laws in Set Theory
  - $(A \cap B)^C = A^C \cup B^C$
  - $(A \cup B)^C = A^C \cap B^C$
- Involutivity of the Complement
  - $(A^C)^C = A$

NB: An involution is a map such that applying it twice gives the identity. Familiar examples: reflecting across the x-axis, the y-axis, or the origin in the plane.

- Transitivity of Inclusion
  - $A \subseteq B \wedge B \subseteq C \rightarrow A \subseteq C$
- Criterion for proving equality of sets
  - $A = B \leftrightarrow A \subseteq C \wedge B \subseteq A$
- Criterion for proving non-equality of sets
  - $A \neq B \leftrightarrow (A \backslash B) \cup (B \backslash A) \neq 0$

4.3 Example Proof in Set Theory

Proposition: $\forall A, B$ sets. $(A \cap B) \cup (A \backslash B) = A$

Proof: Use the criterion for proving equality of sets from above, **i.e.** inclusion in both directions.

Show $(A \cap B) \cup (A \backslash B) \subseteq A$: $\forall x \in (A \cap B) \cup (A \backslash B), x \in (A \cap B)$ or $x \in A \backslash B$. If $x \in (A \cap B)$ then clearly $x \in A$ as $A \cap B \subseteq A$ by definition. If $x \in A \backslash B$, then by definition $x \in A$ and $x \notin B$ so definitely $x \in A$. In both cases, $x \in A$ as needed.

Show $A \subseteq (A \cap B) \cup (A \backslash B)$: $\forall x \in A$, we have two possibilities, namely $x \in B$ or $x \notin B$. If $x \in B$, then $x \in A$ and $x \in B$, so $x \in A \cap B$. If $x \notin B$, then $x \in A$ and $x \notin B$, so $x \notin A \backslash B$. In both cases, $x \in (A \cap B)$ or $x \in (A \backslash B)$ so $x \in (A \cap B) \cup (A \backslash B)$ as needed.

qed

5 The Power Set

Task: Understand what the power set of a set A is.

Definition: Let A be a set. The power set of A denoted $P(A)$ is the collection of all the subsets of A.

Recall: $\emptyset \subseteq A$. It is also clear from the definition of a subset that $A \subseteq A$.

Examples:

1. $A = \{0, 1\}$
   $P(A) = \{\emptyset, \{0\}, \{1\}, \{0, 1\}\}$
2. $A = \{a, b, c\}$
   $P(A) = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, b\}, \{a, c\}, \{b, c\}, \{a, b, c\}\}$
3. $A = \emptyset$
   $P(A) = \{\emptyset\}$
   $P(P(A)) = \{\emptyset, \{\emptyset\}\}$

NB: $\emptyset$ and $\{\emptyset\}$ are different objects. $\emptyset$ has no elements, whereas $\{\emptyset\}$ has one element.

Remark: $P(A)$ and A are viewed as living in separate worlds to avoid phenomena like Russell' paradox.

Q: If A has n elements, how many elements does $P(A)$ have?

A: 2^n

Theorem: Let A be a set with n elements, then $P(A)$ contains 2^n elements.

Proof: Based on the on/off switch idea.

$\forall x \in A$, we have two choices: either we include x in the subset or we don't (on vs off switch). A has n elements $\Rightarrow$ we have 2^n subsets of A.

qed

Alternate Proof: Using mathematical induction.

NB: It is an axiom of set theory (in the ZFC standard system) that every set has a power set, which implies no set consisting of all possible sets could limit, else what would its power set be?

6 Cartesian Products

Task: Understand sets like $\mathbb{R}^1$ in a more theoretical way.

Recall from Calculus:

$\mathbb{R} = \mathbb{R}^1 \ni x$

$\mathbb{R} \times \mathbb{R} = \mathbb{R}^2 \ni (x_1, x_1)$

$\vdots$

$\underbrace{\mathbb{R} \times \mathbb{R}}_{\text{n times}} = \mathbb{R}^2 \ni (x_1, x_2, ..., x_n)$

These are examples of Cartesian products.

Definition: Let A, B be sets. The Cartesian product denoted by $A]timesB$ consists of all ordered pairs $(x, y)stx \in A \wedge y \in B$, **i.e.** $A \times B = \{(x, y) \mid x \in A \wedge y \in B\}$

Further Examples:

1. $A = \{1, 3, 7\}$
   $B = \{1, 5\}$
   $A \times B = \{(1, 1), (1, 5), (3, 1), (3, 5), (7, 1), (7, 5)\}$
   NB: The order in which elements in a pair matters: $(7, 1)$ is different from $(1, 7)$. This is why we call (x, y) an ordered pair.
2. $A = \{(x, y) \in \mathbb{R}^| x^2 + y^2 = 1\} \leftarrow$ circle of radius 1
   $B = \{z \in \mathbb{R} \mid -2 \leq z \leq 2\} = \{-2, 2\} \leftarrow$ closed interval
   $A \times B \leftarrow$ cylinder of radius 1 and height 4

6.1 Cardinality (number of elements) in a Cartesian product

If A has n elements and B has p elements, $A \times B$ has np elements.

Example:

1. $\#(A) = 3 \qquad A = \{1, 3, 7\}$
   $\#(B) = 2 \qquad B = \{1, 5\}$
   $\#(A) \times B) = 3 \times 2 = 6$
2. Both A and B are infinite sets, so $A \times B$ is infinite as well.

Remark: We can define Cartesian products of any length, **e.g.** $A \times A \times B \times A$, $B \times A \times B \times A \times B$, etc. If all sets are finite, the number of elements is the product of the numbers of elements of each factor. If $\#(A) = 3$ and $\#(B) = 2$ as above, $\#(A \times B]timesA) = 3 \times 3 \times 3 = 18$ and $\#(B \times A \times B) = 2 \times 3 \times 2 = 12$.

7 Relations

Task: Define subsets of Cartesian products with certain properties. Understand the predicates " = " (equality) and other predicates in predicate logic in a more abstract light.

Start with $x = y$. The elements x is some notation R to y (equality in this case). We can also denote it as xRy or $(x, y) \in E$)

Let x, y in $\mathbb{R}$, then $E = \{(x, x) \mid x \in \mathbb{R}\} \subset \mathbb{R} \times \mathbb{R}$.

The "diagonal" in $\mathbb{R} \times \mathbb{R}$ gives exactly the elements equal to each other.

More generally:

Definition: Let A, B be sets. A subset of the Cartesian product $A \times B$ is called a relations between A and B. A subset of the Cartesian product $A \times A$ is called a relations on A.

Remark: Note how general this definition is. To make it useful for understanding predicates, we will need to introduce key properties relations can satisfy.

Example: $A = \{1, 3, 7\} \qquad B = \{1, 2, 5\}$

We can define a relation S on $A \times B$ by $S = \{(1, 1), (1, 5), (3, 2)\}$. This means $1S1$, $1S5$ and $3S2$ and no other ordered pairs in $A \times B$ satisfy S.

Remark: The relations we defined involve 2 elements, so they are often called binary relations in the literature.

8 Equivalence Relations

Task: Define the most useful kind of relation.

Definition: A relation R on a set A is called

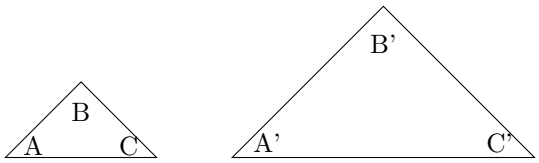
1. reflexive iff (if and only if) $\forall x \in A, xRx$
2. symmetric iff $\forall x, y \in A, xRy \rightarrow yRx$
3. transitive iff $\forall x, y, z \in A, xRy \wedge yRz \rightarrow xRz$

An equivalence relation on A is a relation that is reflexive, symmetric and transitive.

Notation: Instead of xRy, an equivalence relation is often denoted by $x \equiv y$ or $x \sim y$.

Examples:

1. "=" equality is an equivalence relation.
   (a) $x = x$ reflexive
   (b) $x = y \Rightarrow y = x$ symmetric
   (c) $x = y \wedge y = z \Rightarrow x = z$ transitive
2. $A = \mathbb{N}$
   $x \equiv y \bmod 3$ is an equivalence relation. $x \equiv y \bmod 3$ means $x - y = 3m$ for some $m \in \mathbb{Z}$, **i.e.** x and y have the same remainder when divided by 3. The set of all possible remainders is $\{0, 1, 2\}$
   NB: In correct logic notation, $x \equiv y \bmod 23$ if $\exists m \in \mathbb{Z} s.t. x - y = 3m$
   (a) $x \equiv x \bmod 3$ since $x - x = 0 = 3 \times 0 \rightarrow$ reflexive
   (b) $x \equiv y \bmod 3 \Rightarrow y \equiv x \bmod 3$ because $x \equiv y \bmod 3$ means $x - y = 3m$ for some $m \in \mathbb{Z} \Rightarrow y - x = -3m = 3 \times (-m) \Rightarrow y \equiv x \bmod 3 \rightarrow$ symmetric
   (c) Assume $x \equiv y \bmod 3$ and $y \equiv z \bmod 3$
   $x \equiv y \bmod 3 \Rightarrow \exists m \in \mathbb{Z}$ s.t. $x - y = 3m \Rightarrow y = x - 3m$
   $y \equiv z \bmod 3 \Rightarrow \exists p \in \mathbb{Z}$ s.t. $y - z = 3p \Rightarrow y = z + 3p$
   Therefore, $x - 3m = z + 3p \Leftrightarrow x - z = 3p + 3m = 3(p + m)$
   Since $p, m \in \mathbb{Z}, p + m \in \mathbb{Z} \Rightarrow x \equiv z \bmod 3 \rightarrow$ transitive.
3. Let $f : A \rightarrow A$ be any function on a non empty set A. We define the relation $R = \{(x, y) \mid f(x) = f(y)\}$
   (a) $\forall x \in A, f(x) = f(x) \Rightarrow (x, x) \in R \rightarrow reflexive$
   (b) If $(x, y) \in R$, then $f(x) = f(y) \Rightarrow f(y) = f(x)$, **i.e.** $(y, x) \in R \rightarrow$ symmetric
   (c) If $(x, y) \in R$ and $(y, z) \in R$, then $f(x) = f(y)$ and $f(y) = f(z)$, which by the transitivity of equality implies $f(x) = f(z)$, **i.e.** $(x, z) \in R$ as needed, so R is transitive as well.
   $f(x)$ can be $e^x, sin\, x, (x)$, etc.



4. Let λ be the set of all triangles in the plane. $ABC \sim A'B'C'$ if ABC and $A'B'C'$ are similar triangles, **i.e.** have equal angles.
   (a) $\forall ABC \in \lambda, ABC \sim ABC$ so $\sim$ is reflexive
   (b) $ABC \sim A'B'C' \Rightarrow A'B'C' \sim ABC$ so $\sim$ is symmetric
   (c) $ABC \sim A'B'C'$ and $A'B'C' \sim A"B"C" \Rightarrow ABC \sim A"B"C"$, so $\sim$ is transitive

   Clearly (a), (b), (c) use the fact that equality of angles is an equivalence relation.

Exercise: For various predicates you've encountered, check whether reflexive, symmetric or transitive. Examples of predicates include $\neq, <, >, \leq, \geq, \subseteq$

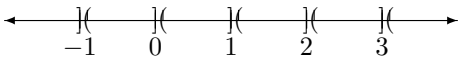
9 Equivalence Relations and Partitions

Task: Understand how equivalence relations divide sets.

Definition: Let A be a set. A partition of A is a collection of non empty sets, any two of which are disjoint such that their union is A, **i.e.** $\lambda = \{A_\alpha \mid \alpha \in I\}$ s.t. $\forall \alpha, \alpha' \in I$ satisfy $\alpha \neq \alpha', A_\alpha \cap A_{\alpha'} \neq \emptyset$ and $\bigcup\limits_{\alpha \in I} A_\alpha = A$

Here I is an indexing act (may be infinite). $\bigcup\limits_{\alpha \in I} A_\alpha$ is the union of all the A_α's (possibly an infinite union)

Example $\{(n, n+1) \mid n \in \mathbb{Z}\}$ is a partition of $\mathbb{R}$



$\bigcup\limits_{n \in \mathbb{Z}} (n, n+1] = \mathbb{R}$

$(n, n+1] \cap (m, m+1] = \emptyset$ if $n \neq m$

Definition: If R is an equivalence relations on a set A and $x \in A$, the equivalence class of x denoted $[x]_R$ is the set $\{y \mid xRy\}$. The collection of all equivalence classes is called A modulo R and denoted A/R.

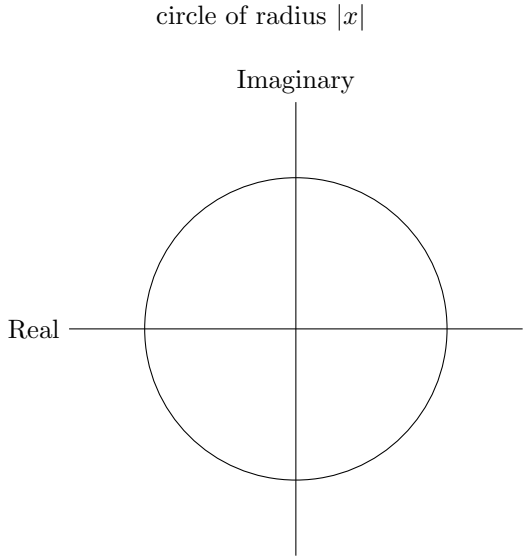
Examples:

1. $A = \mathbb{N}$ $\quad x \equiv y \bmod 3$
   We have the equivalence classes $[0]_R, [1]_R$ and $[2]_R$ given by the then possible remainders under division by 3.
   $[0]_R = \{0, 3, 6, 9, ...\}$
   $[1]_R = \{1, 4, 7, 10, ...\}$
   $[2]_R = \{2, 5, 8, 11, ...\}$
   Clearly $[0]_R \cup [1]_R \cup [2]_R = \mathbb{N}$ and they are mutually disjoint $\Rightarrow R$ gives a partition of $\mathbb{N}$.

2. $ABC \sim A'B'C'$
   $[ABC] = \{$The set of all triangles with angles of magnitude $\measuredangle ABC, \measuredangle BAC, \measuredangle ACB\}$
   The union over the set of all $[ABC]$ is the set of all triangles and $[ABC] \cap [A'B''] = \emptyset$ if $ABC \neq^* A'B'C'$ since it means these triangles have at least one angle that if difference.

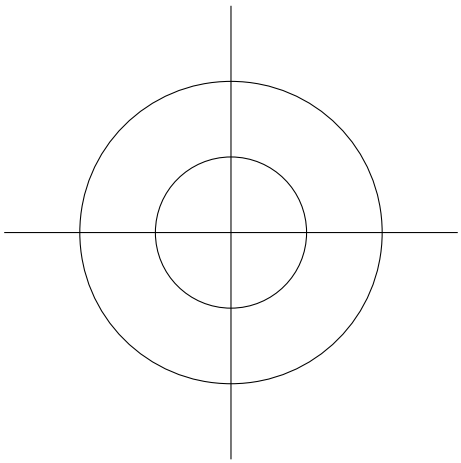
   * In the original notes, not $\sim$ is used (a tilde with a slash going through it) but I couldn't find this symbol in latex.

3. $A = \mathbb{C}$ $\quad x \cap y$ if $|x| = |y|$ $\quad$ equivalence relation
   $[x] = \{y \in \mathbb{C} \mid |x| = |y|\} = [r]$ for $r \in [0, +\infty) \wedge (r \geq 0)$



   $\bigcup_{r \in [0,+\infty)} [r] = \mathbb{C}$
   $[r_1] \cap [r_2] \neq \emptyset$ if $r_1 \neq r_2$ since two distinct circles in $\mathbb{C} \simeq \mathbb{R}^2$ with empty intersection.

circles $r_1 \wedge r_2$



Theorem: For any equivalence relation R on a set A, its equivalence classes form a partition of A, **i.e.**

1. $\forall x \in A, \exists y \in A$ s.t. $x \in [y]$ (every element of A sits somewhere)
2. $xRy \Leftrightarrow [x] = [y]$ (all elements related by R belong to the same equivalence class)
3. $\neg(xRy) \Leftrightarrow [x] \cap [y] = \emptyset$ (if two elements are not related by R, the they belong to disjoint equivalence classes)

Proof:

1. Trivial. Let $y = x$. $x \in [x]$ because R is an equivalence relation. Hence reflexive, so xRx holds.
2. We will prove $xRy \Leftrightarrow [x] \subseteq [y]$ and $[y] \subseteq [x]$
   $\Rightarrow$ Fix $x \in A, [x] = \{z \in A \mid xRz\} \Rightarrow \forall y \in A$ s.t. $xRy, y \in [x]$. Furthermore, $[y] = \{w \in A \mid yRw\}$
   $\Rightarrow \forall w \in [u], yRw$ but $xRy \Rightarrow xRw$ by transitivity. Therefore, $w \in [x]$. We have shown $[y] \subseteq [x]$.
   Since R is an equivalence relation, it is also symmetric. **i.e.** $xRy \Leftrightarrow yRx$. So by the same argument with x and y swapped $yRx \Rightarrow [x] \subseteq [y]$. Thus $xRy \Rightarrow [x] = [y]$.
   $\Rightarrow [x] = [y] \Rightarrow y \in [x]$ but $[x] = \{y \in A \mid xRy\}$
3. $\Rightarrow$ We will prove the contrapositive. Assume $[x] \cap [y] \neq \emptyset \Rightarrow \exists z \in [x] \cap [y].z \in [x]$ means xRy, whereas $z \in [y]$ means $yRx \Leftrightarrow zRy$ by symmetric of R. We thus have xRz and $zRy \Rightarrow xRy$ by transitivity of $R.xRy$ contradicts $\neg(xRy)$ so indeed $\neg(xRy) \Rightarrow [x] \cap [y] = \emptyset$
   $\Leftarrow$ Once again we use the contrapositive.
   Assume $\neg(\neg(xRy)) \Leftrightarrow xRy$. By part (b) $xRy \Rightarrow [x] = [y] \Rightarrow [x] \cap$

$[y] \neq \emptyset$ since $x \in [x]$ and $y \in [y]$, **i.e.** These equivalence classes are non empty. We have obtained the needed contradiction.

qed

Q: What partition does "=" impose on $\mathbb{R}$?

A: $[x] = \{x\}$ since $E = \{(x, x) \mid x \in \mathbb{R}\}$ the diagonal.
The one element equivalence class is the smallest equivalence class possible (by definition, an equivalence class cannot be empty as it contains x itself). We call such a partition the finest possible partition.

Remark: The theorem above shows how every equivalence relations partitions a set. It turns out every partition of a set can be used to define an equivalence relation: xRy is x nd y belong to the same subset of the partition (check this is indeed an equivalence relations!). Therefore, there is a 1-1 correspondence between partitions and equivalence relations: to each equivalence relation there corresponds a partition and vice versa.

10 Partial Orders

Task: Understand another type of relation with special properties.

Definition: Let A be a set. A relation R on A is called anti-symmetric if $\forall x, y \in A$ s.t. $xRy \wedge yRx$, then $x = y$.

Definition: A partial order is a relation on a set A that is reflexive, anti-symmetric, and transitive.

Examples:

1. $A = \mathbb{R}$ $\leq$ "less than or equal to" is a partial order
   (a) $\forall x \in \mathbb{R} x \leq x \rightarrow$ reflexive
   (b) $\forall x, y \in \mathbb{R}$ s.t. $x \leq y \wedge y \leq x \implies x = y \rightarrow$ anti-symmetric
   (c) $\forall x, y, z \in \mathbb{R}$ s.t. $x \leq y \wedge y \leq z \implies x \leq z \rightarrow$ transitive
   Same conclusion if $A = \mathbb{Z} \vee A\mathbb{N}$
2. A is a set. Consider $P(A)$, the power set of A. The relation $\subseteq$ "being a subset of" is a partial order.
   (a) $\forall B \in P(A), B \subseteq B \rightarrow$ reflexive.
   (b) $forallB, C \in P(A), B \subseteq C \wedge C \subseteq B \implies B = C$ (recall the criterion for proving equality of sets) $\rightarrow$ anti-symmetric
   (c) $\forall B, C, D \in P(A)$ s.t. $B \subseteq C \wedge C \subseteq D \implies B \subseteq B \rightarrow$ transitive

The most important example of a partial order is example (2) "being a subset of".

Q: Why is "being a subset of" a partial order as opposed to a total order?

A: There might exist products B, C of A s.t. neither $B \subseteq C$ nor $C \subseteq B$ holds, **i.e.** where $B \wedge C$ are not related via inclusion.

11 Functions

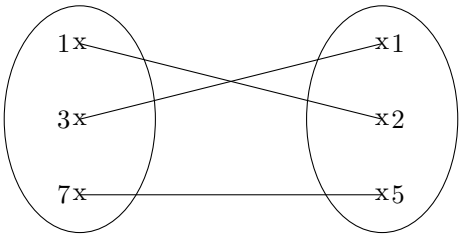
Task: Define a function rigorously and make sense of terminology associated to functions.

Definition: Let A, B be sets. A function $f : A \to B$ is a rule that assigns to every element of A one and only one elements of B, **i.e.** $\forall x \in A \exists! y \in B$ s.t. $f(x) = y$. A is called the domain of f and B is called the codomain.

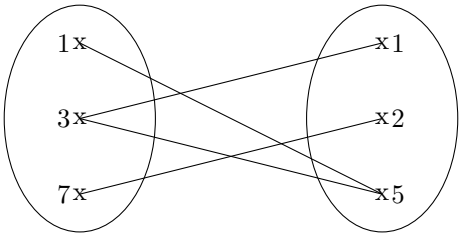
Examples:

1. $A = \{1, 3, 7\}$
   $B = \{1, 2, 5\}$

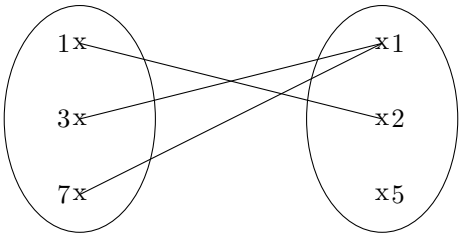
Is a function.



Not a function; 3 sent to both 1 and 5



Is a function.



2. $A = B = \mathbb{R}$ $F : \mathbb{R} \to \mathbb{R}$ given by $f(x) = x$ is called the identity function.

Definition: Let A, B be sets and let $f : A \to B$ be a function. The range of f denoted by $f(A)$ if the subset of B defined by $f(A) = \{y \in B \mid \exists x \in A$ s.t. $f(x) = y\}$.

Definition: Let A be a set. A Boolean function on A is a function $F : A \to \{T, F\}$ which has A as its domain and the set of truth values $\{T, F\}$ as is codomain. $f : A \to \{T, F\}$ thus assigns truth values to the elements of A.

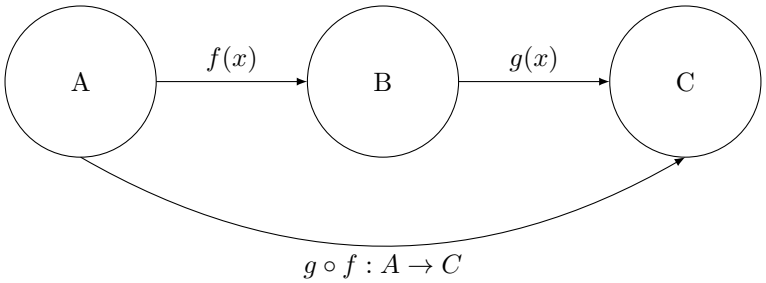
Function are often represented by graphs. If $f : A \to B$ is a function, the graph of f denoted $\Gamma(f)$ is the subset of the Cartesian product $A \times B$ given by $\{(x, f(x)) \mid x \in A\}$.

Q: Is it possible to obtain every subset of $A \times B$ as the graph of some function?

A: No! For $f : A \to B$ to be a function $\forall x] in A \exists! y \in B$ s.t. $f(x) = y$, so for $\Gamma \subseteq A \times B$ to be the graph of some function, Γ must satisfy that $\forall x \in A \exists! y \in B$ s.t. $(x, y) \in \Gamma$. Then we can define f by letting $y = f(x)$.

12 Composition of Functions

Task: Understand the natural operation that allows us to combine functions.



Example:

$f : \mathbb{R} \to \mathbb{R}$ $\quad f(x) = 2x$
$g : \mathbb{R} \to \mathbb{R}$ $\quad cos\ x$
$g \circ f(x) = g(f(x)) = g(2x = cos(2x))$
$f \circ g(x) = f(g(x)) = f(cos\ x) = 2(cos\ x) = 2cos\ x$

13 Inverting Functions

Task: Figure out which properties a function has to satisfy so that its action can be undone, **i.e.** when we can define an inverse to the original function.

Given $f : A \to B$, want $f^{-1} : B \to A$ s.t. $f^{-1} \circ f : A \to A$ is the identity
$f^{-1} \circ f(x) = f^{-1}(f(x)) = x$
$A \overset{f}{\to} B \overset{f^{-1}}{\to} A$
It turns out f has to satisfy two properties for f^{-1} to exist.

1. Injective

2. Surjective

Definition: A function $f : A \to B$ is called injective or an injection (sometimes called one to one) if $f(x) = f(y) \Rightarrow x = y$

Examples:
$\sin x : [0, \frac{\pi}{2}] \to \mathbb{R}$ is injective
$\sin x : \mathbb{R} \to \mathbb{R}$ is not injective because $\sin x = \sin \pi = 0$

Definition: A function $f : A \to B$ is called surjective or a surjection (sometimes called onto) if $\forall z \in B \exists x \in A$ s.t. $f(x) = z$.

Remark: f assigns a value to each element of A by its definition as a function, but it is not required to cover all of B. f is surjective if its range is all of B.

Examples:
$\sin x : \mathbb{R} \to [-1, 1]$ is surjective
$\sin x : \mathbb{R} \to \mathbb{R}$ is not surjective since $\nexists x \in \mathbb{R}$ s.t. $\sin x = 2$. We know $|\sin x| \leq 1 \forall x \in \mathbb{R}$

Definition: A function $f : A \to B$ is called bijective or a bijection if f is both injective and surjective.

Example: $f : \mathbb{R} \to \mathbb{R}$ $\quad f(x) = 2x + 1$ is bijective.

- Check injectivity $f(x_1) = f(x_2) \Rightarrow 2x_1 + 1 = 2x_2 + 1 \Leftrightarrow 2x_1 = 2x_2 \Leftrightarrow x_1 = x_2$ as needed.
- Check surjectivity $\forall z \in \mathbb{R}. f(x) = z$ means $2x + 1 = z$.
  Solve for x: $2x = z - 1 \Rightarrow x = \frac{z-1}{2} \in \mathbb{R} \Rightarrow f$ is surjective.

Remark: All bijective functions have inverses because we can define the inverse of a bijection and it will be a function:

- Surjectivity ensures f_{-1} assigns an element to every element of B (its domain).
- Injectivity ensures f_{-1} assigns to each elements of B one and only one elements of A.

Conclusion: $f : A \to B$ bijective $\Rightarrow$ f_{-1} exists, **i.e.** f_{-1} is a function. It turns out (reverse the arguments above) that f_{-1} exists $\Rightarrow f : A \to B$ is bijective.
Altogether we get the following theorem:

Theorem: Let $f : A \to B$ be a function. f_{-1} exists $\Leftrightarrow f : A \to B$ is bijective.

Q: How do we find the inverse function f_{-1} given $f : A \to B$?

A: If $f(x) = y$, solve for x as a function of y since $f_{-1}(f(x)) = f_{-1}(y) = x$ s $f_{-1} \circ f$ is the identity.

Example: $f(x) = 2x + 1 = y$. Solce for x in terms of y.
$f : \mathbb{R} \to \mathbb{R}$
$2x = y - 1 \qquad x = \frac{y-1}{2}$

14 Functions Defined on Finite Sets

Task: Derive conclusions about a function given the number of elements of the domain and codomain if finite; understand the pigeonhole principle.

Proposition: Let A, B be sets and let $f : A \to B$ be a function. Assume A is finite. Then f is injective $\Leftrightarrow$ $f(A)$ has the same number of elements as A.

Proof:

A is finite so we can write it as $A = \{a_1, a_2, ..., a_p\}$ for some p. Then $f(A) = \{f(a_1), f(a_2), ..., f(a_p)\} \subseteq B$. A priori, some $f(a_i)$ might be the same as some $f(a_j)$. However, f injective $\Leftrightarrow$ $f(a_i) \neq f(a_j)$ whenever $i \neq j \Leftrightarrow f(A)$ has example p elements just like A.

qed

Corollary 1 Let A, B be finite sets such that $\#(A) = \#(B)$. Let $f : A \to B$ be a function. f is injective $\Leftrightarrow$ f is bijective.

Proof:

$\Rightarrow$ Suppose $f : A \to B$ is injective. Since A is finite, by the previous proposition, $f(A)$ has the same number of elements as A, but $f(A) \subseteq B$ and B has the same number of elements as $A \Rightarrow \#(A) = \#(f(A)) = \#(B)$, which means $f(A) - B$, **i.e.** f is also surjective $\Rightarrow$ f is bijective.

$\Leftarrow$ f is bijective $\Leftarrow$ f is injective.

qed

Corollary 2 (The Pigeonhole Principle) Let A, B be finite sets. If $\#(B) < \#(A)$, and let $f : A \to B$ be a function. $\exists a, a' \in A$ where $a \neq a'.f(a) = f(a')$

Remark: The name pigeonhole principle is due to Paul Erdös and Richard Rado. Before is was known as the principle of the drawers of Dirichlet. It has a simple statement, but it's a very powerful result in both mathematics and computer science.

Proof: Since $f(A) \subseteq B$ and $\#(B) < \#(A)$, $f(A)$ cannot hve as many elements as A, so by the proposition, f cannot be injective, **i.e.** $\exists a, a' \in A$ where $a \neq a'$ (**i.e.** distinct elements) s.t. $f(a) = f(a')$

qed

Examples:

1. You have 8 friends. At least two of them were born the same day of the week. #(days of the week) = 7 < 8.
2. A family of five fives each other presents for Christmas. There are 12 presents under the tree. We conclude at least one person for three presents or more.
3. In a list of 30 words in English, at least two will begin with the same letter. #(Letter in the English alphabet) = 26 ¡ 30.

14.1 Behaviour of Functions on Infinite Sets

Let A be a set and $f : A \to A$ be a function. If A is finite , the corollary 1 tells us f injective $\Leftrightarrow$ f bijective. What if A is not finite?

14.1.1 Hilbert's Hotel problem (jazzier name: Hilbert's paradox of the Grand Hotel)

A fully occupied hotel with infinitely many rooms can always accommodate an additional guest as follows: The person in Room 1 removes to Room 2. The person in Room 2 moves to Room 3 and so on, **i.e.** if the rooms at $x_1, x_2, x_3...$ define the function $f(x_1) = x_2, f(x_2) = x_3, ..., f(x_m) = x_{m+1}$.

Claim: As defined f is injective but not surjective (hence not bijective!). Let $H = \{x_1, x_2, ...\}$ the hotel consisting of infinitely many rooms. $f : H \to H$ is given by $f(x_n) = f(x_{n+1})$. $f(H) = H \backslash \{x_1\}$. We cn use this idea to prove:

Proposition: A set A is finite $\Leftrightarrow$ $\forall f : A \to A$ an injective function is also bijective.

Proof: If the set X is finite then it follows immediately that every injective function $f : X \to X$ is bijective.
Suppose that the set X is infinite. Then there exists some infinite sequence $x_1, x_2, x_3, ...$ of distinct elements of X (where an element of X occurs at most once in this list). Then there exists a function $f : X \to X$ defined such that $f(x_n) = x_{n+1}$ for all positive integers of n, and $f(x) = x$ for all elements of x of X. If x is not a member of the infinite sequence $x_1, x_2, x_3, ...$ then the only elements of X that gets mapped to x is the element x itself; if $x = x_n$, where $x > 1$, then the only element of X gets mapped to x. It follows that the function f is injective. However it is not surjective, since x_1 does not belong to the range of the function. This function f is thus an example of a function from the set X to itself which is injective but not bijective.

15 Mathematical Induction

Task: Understand how to construct a proof using mathematical induction.

$\mathbb{N} = \{0, 1, 2, ...\}$ set of natural numbers.
Recall that $\mathbb{N}$ is constructed using 2 axioms:

1. $0 \in \mathbb{N}$
2. If $n \in \mathbb{N}$, then $n + 1 \in \mathbb{N}$

Remarks:

1. This is exactly the process of counting.
2. If we start at 1, then we construct $\mathbb{N}* = \{1, 2, 3, 4, ...\} = \mathbb{N}\backslash\{0\}$

via the axioms

1. $1 \in \mathbb{N}*$
2. if $n \in \mathbb{N}*$, then $n + 1 \in \mathbb{N}*$

$\mathbb{N}$ or $\mathbb{N}*$ is used for mathematical induction.

15.1 Mathematical Induction Consists of Two Steps:

Step 1 Prove statements $P(1)$ called the base case.

Step 2 For any n, assume $P(n)$ and prove $P(n+1)$. This is called the inductive step.
In other words, step 2 proves the statement $\forall n P(n) \to P(n + 1)$

Remark: Step 2 is not just an implication but infinitely many! In logic notation, we have:

Step 1 $P(1)$

Step 2 $\forall n(P(n) \to P(n + 1))$
Therefore, $\forall n P(n)$
Let's see how the argument proceeds:

1. $P(1)$ Step 1 (base case)
2. $P(1) \to P(2)$ by Step 2 with $n = 1$
3. $P(2)$ by 1 & 2
4. $P(2) \to P(3)$ by Step 2 with $n = 2$
5. $P(3)$ by 3 & 4
6. $P(3) \to P(4)$ by Step 2 with $n = 3$
7. $P(4)$ by 5 & 6

$\vdots$

8. $P(n)$ for any n.
   This is like a row of dominos: knocking over the first one in a row makes all the others fall. Another idea is climbing a ladder.

Examples:

1. Prove $1+3+5+...+(2n-1)=n^2$ by induction.

   Base Case: Verify statement for $n=1$

   When $n=1$, $2n-1=2\times 1-1=1^2$

   Inductive Step: Assume $P(n)$, **i.e.** $1+3+5+...+(2n-1)=n^2$ and seek to prove $P(n+1)$, **i.e.** the statement $1+3+5...+(2n-1+2(n+1)-1)=(n+1)^2$

   We start with LHS: $1+\underbrace{3+5+...+(2n-1)}_{n^2}+(2(n+1)-1)=$

   $n^2+2n+2-1=n^2+2n+1=(n+1)^2$

2. Prove $1+2+3++n=\frac{n(n+1)}{2}$ by induction.

   Base Case: Verify statement for $n=1$

   When $n=1, 1=\frac{1\times(1+1)}{2}=\frac{1\times 2}{2}=1$

   Inductive Step: Assume $P(n)$, **i.e.** $1+2+3+...+n=\frac{n\times(n+1)}{2}$ and seek to prove $1+2+3+...+n=\frac{(n+1)(n+2)}{2}$

   $\underbrace{1+2+3+...+n}_{\frac{n(n+1)}{2}}+n+1=\frac{n(n+1)}{2}+n+1=(n+1)(\frac{n}{2}+1)=$

   $(n+1)\frac{n+2}{2}=\frac{(n+1)(n+2)}{2}$ as needed.

Remarks:

1. For some argument by induction, it might be necessary to assume not just $P(n)$ at the inductive step but also $P(1), P(2)P, ...mP(n-1)$. This is called strong induction.

   Base Case: Prove $P(1)$

   Inductive Step: Assume $P(a), P(2), ..., P(n)$ and prove $P(n+1)$.

   An example of result requiring the use of strong induction is the Fundamental Theorem of Arithmetic: $\forall n\in\mathbb{N}, n\geq 2, n$ can be expressed as a product of one or more prime numbers.

2. One has to be careful with argument involving induction. Here is an illustration why:

   Polya's argument that all horses are the same colour:

   Base Case: $P(1)$ There is only one horse, sot hat has a colour.

   Inductive Step Assume any n horses are the same colour.

   Consider a group of $n+1$ horses. Exclude the first horse and look at the other n. All of these are the same colour by our assumption. Now exclude the last horse. The remaining n horses are the same

colour by our assumption. Therefore, the first horse, the horses in the middle, and the last horse are all of the same colour. We have established the inductive step.

Q: Where does the argument fail?

A: For $n = 2, P(2)$ is false because there are no middle horses to compare to.

item[] The Grand Hotel Cigar Mystery

Recall Hilbert's hotel - the grand Hotel. Suppose that the Grand Hotel does not allow smoking and no cigars may be taken into the hotel. In spite of the rules, the guest in Room 1 goes to Room 2 to get a cigar. The guest in Room 2 goes to Room 3 to get 2 cigars (one for him and one for the person in room 1), etc. In other words, guest in Room N goes to Room N+1 to get N cigars. They will each get back to their rooms, smoke one cigar, and give the result to the person in Room N-2.

Q: Where is the fallacy?

A: This is an induction argument without a base case. No cigars are allowed in the hotel so no guests have cigars. An induction cannot get off the ground without a base case.

16 Abstract Algebra

Task: Understand binary operators, semigroups, monoids, and groups as well as their properties.

16.1 Binary Operations

Definition: Let A be a set. A binary operation $*$ on A is an operation applied to any two elements $x, y \in A$ that yields on elements $x * y$ in A. In other words, $*$ us a binary operation on A if $\forall x, y \in A, x * y \in A$.

Examples:

1. $\mathbb{R}, +$ addition on $\mathbb{R} : \forall x, y \in \mathbb{R}, x + y \in \mathbb{R}$
2. $\mathbb{R}, -$ subtraction on $\mathbb{R} : \forall x, y \in \mathbb{R}, x - y \in \mathbb{R}$
3. $\mathbb{R}, \times$ multiplication on $\mathbb{R} : \forall x, y \in \mathbb{R}, x \times y \in \mathbb{R}$
4. $\mathbb{R}, /$, division on $\mathbb{R}$ is NOT a binary operation because $\forall x \in \mathbb{R} \exists o \in \mathbb{R}$ s.t. $\frac{x}{o}$ is undefined (not an element of $\mathbb{R}$)

Definition: A binary operation $*$ on a set A is called commutative if $\forall x, y \in A, x * y = y * x$

Examples:

1. $\mathbb{R}, +$ is commutative since $\forall x, y \in \mathbb{R}, x + y = y + x$

2. $\mathbb{R}, \times$ is commutative since $\forall x, y \in \mathbb{R}, x \times y = y \times x$
3. $\mathbb{R}, -$ is not commutative since $\forall x, y \in \mathbb{R}, x - y \neq y - x$ in general. $x - y = y - x$ only if $x = y$
4. Let Mn be the set of n by n matrices with entries in $\mathbb{R}$ and let $*$ be matrix multiplication. $\forall A, B \in Mn, A * B \in Mn$, so $*$ is a binary operation, but $AB \neq BA$ in general. Therefore $*$ is not commutative.

Definition: A binary operation $*$ on a set A is called <u>associative</u> if $\forall x, yz(x * y) * z = x * (y * z)$

Examples:

1. $\mathbb{R}, +$ is associative since $\forall x, y, z \in \mathbb{R}, (x + y) + z = x + (y + z)$
2. $\mathbb{R}, \times$ is associative since $\forall x, y, z \in \mathbb{R}, (x \times y) \times z = x \times (y \times z)$
3. Intersection $\cap$ on sets is associative since $\forall A, B, C$ sets $(A \cap B) \cap C = A \cap (B \cap C)$.
4. Union $\cup$ on sets is associative since $\forall A, B, C$ sets $(A \cup B) \cup C = A \cup (B \cup C)$
5. $\mathbb{R}, -$ is not associative since $(1-3)-5 = -2-5 = -7$ but $1-(3-5) = 1-(-2) = 1+2 = 3$

Remark: When we are dealing with associative binary operations we can drop the parentheses, **i.e.** $(x * y) * z$ can be written $x * y * z$.

16.2 Semigroups

Definition: A <u>semigroup</u> is a set endowed with an associative binary operation. We denote the semigroup $(A, *)$

Examples:

1. $(\mathbb{R}, +)$ and $(\mathbb{R}, -)$ are semigroups.
2. Let A be a set and let $P(A)$ be its power set. $(P(A), \cap)$ and $(P(A), \cup)$ are both semigorups.
3. $(Mn, *)$, the set of $n \times n$ matrices with entries in $\mathbb{R}$ with the operation of matrix multiplication (which is associative $\rightarrow$ a bit gory to prove) forms a semigroup.

   Since $*$ is associative on a semigroup, we can define a^n :
   $a^1 = a$
   $a^2 = a * a$
   $a^3 = a * a * a$
   $\vdots$
   Recursively, $a^1 = 1$ and $a^n = a * a^{n-1}, \forall n > 1$

NB: In $(\mathbb{R}, \times), \forall a \in \mathbb{R}, a^n = \underbrace{a \times a \times ... \times a}_{n\ times}$, whereas in $(\mathbb{R}, +), \forall a \in \mathbb{R}, a^n = \underbrace{a + a + ... + a}_{n\ times} = na$. Be careful what $*$ stands for!

Theorem: Let $(A, *)$ be a semigroup. $\forall a \in A, a^m * a^n = a^{m+n}, \forall m, n \in \mathbb{N}*$.

Proof: By induction on m.

Base Case: $m = 1$ $\quad a^1 * a^n = a * a^n = a * 1 + n$

Inductive Step: Assume the result is true for $m = p$, **i.e.** $a^p * a^n = a^{p+n}$ and seek to prove that $a^{p+1} * a^n = a^{p+1+n}$

$a^p + 1 * a^n = (a * a^p) * a^n = a * (a^p * a * n) = a * a^{p+n} = a^{p+1+n}$

Theorem: Let$(A, *)$ be a semigroup. $\forall a \in A, (a^m)^n = a^(mn), \forall m, n \in \mathbb{N}*$

Proof: By induction on n.

Base Case: $n = 1$ $\quad (a^m)^1 = a^m = a^{m \times 1}$

Inductive Step: Assume the result if true for $n = p$, **i.e.** $(a^m)^p = a^{mp}$ and seek to prove that $(a^m)^{p+1} = a^{m(p+1)}$

$(a^m)^{p+1} = (a^m)^p * a^m = a^{mp} * a^m = a^{mp+m} = a^{m(p+1)}$

16.2.1 General Associative Law

Let $(A, *)$ be a semigroup. $\forall a_1, ..., a_s \in A, a_1 * a_2 * ... * a_s$ has the same value regardless of how the product is bracketed.

Proof Use associativity of $*$.

qed

NB: Unless $(A, *)$ has a commutative binary operation, $a_1 * a_2 * ... * a_s$ does depend on the <u>ORDER</u> in which the $a'_j s$ appearin $a_1 * a_2 * ... * a_s$

16.2.2 Identity Elements

Definition: Let $(A, *)$ be a semigroup. An element $e \in A$ is called an identity element for the binary operation $*$ if $e * x = x * e = x, \forall x \in A$.

Examples:

1. $(\mathbb{R}, +)$ has 0 as the identity element.
2. $(\mathbb{R}, \times)$ has 1 as the identity element.
3. Given a set $A, (P(A), \cup)$ has $\emptyset$ (the empty set) as its identity elements, whereas $(P(A), \cap)$ does <u>NOT</u> have an identity element.
4. $(Mn, *)$ has In, the identity matrix as its identity element.

Theorem A binary operation on a set cannot have more than one identity elements, **i.e.** if an identity element exists, then it is unique.

Proof: Assume not (proof by contradiction). Let e and e' both be identity elements for a binary operation on a set A. $e = e * e' = e'$

qed

16.3 Monoids

Definition: A monoid is a set A endowed with an associative binary operation $*$ that has an identity element e. In other words, a monoid is a semigroup $(A, *)$ where $*$ has an identity element e.

Definition: A monoid $(A, *)$ is called commutative (or Abelian) if the binary operation $*$ is commutative.

Example:

1. $(\mathbb{R}, +)$ is a commutative monoid with $e = o$.
2. $(\mathbb{R}, \times)$ is a commutative monoid with $e = 1$.
3. Given a set A, $(P(A), \cup)$ is a commutative monoid with $e = \emptyset$.
4. $(M, n*)$ is a monoid since $e = In$, but it is not commutative since matrix multiplication is not commutative.
5. $(\mathbb{N}, +)$ is a commutative monoid with $e = o$, whereas $(\mathbb{N}*, +)$ is merely a semigroup (recall $\mathbb{N}* = \mathbb{N}\backslash\{0\}$)

Theorem: Let $(A, *)$ be a monoid and let $a \in A$. Then $a^m * a^n = a^{m+n}, \forall m, n \in \mathbb{N}$

Remark: Recall that we proved this theorem for semigroups if $m, n \in \mathbb{N}*$. We now need to extend that result.

Proof: A monoid is a semigroup $\implies \forall a \in A, a^m * a^n = a^{m+n}$ whenever $m, n \in \mathbb{N}*$, **i.e.** $m > 0$ and $n > 0$. Now let $m = 0$. $a^m * a^n = a^0 * a^n = e * a^n = a^n = a^{0+n}$
If $n = 0, a^m * a^n = a^m * a^0 = a^m * e = a^m = a^{m+0}$

qed

Theorem: Let $A, *)$ be a monoid, $\forall a \in A \forall m, n \in \mathbb{N}, (a^m)^n = a^{mn}$

Remark: Once again, we had this result for semigroups when $m > 0$ and $n > 0$

Proof: By the remark, we only need to prove the result when $m = 0$ or $n = 0$. If $m = 0, (a^0)^n = (e)^n = e = a^0 = a^{0 \times n}$. If $n = 0$ then $(a^m)^0 = e = a^0 = a^{0 \times m}$

17 Inverses

Task: Understand what an inverse is and what formal properties it satisfies.

Definition: Let $(A, *)$ be a monoid with identity element e and let $a \in A$. An element y of A is called the inverse of x if $x * y = y * x = e$. If an element $a \in A$ has an inverse, then a is called invertible.

Examples:

1. $(\mathbb{R}, +)$ has identity element 0. $\forall x \in \mathbb{R}, (-x)$ is the inverse of x since $x + (-x) = (-x) = x = 0$.
2. $(\mathbb{R}, \times)$ has identity element 1. $x \in \mathbb{R}$ is invertible only if $x \neq 0$. If $x \neq 0$, the inverse of x is $\frac{1}{x}$ since $x \times \frac{1}{x} = \frac{1}{x} \times x = 1$.
3. $(Mn, *)$ the identity element is In. $A \in Mn$ is invertible if $det(A) \neq 0$. A^{-1} the inverse is exactly the one you computed in linear algebra. If $det(a) = 0, A$ is NOT invertible.
4. Given a set $A, (P(A), \cup)$ has $\emptyset$ as its identity element of all the elements of $P(A)$ only $\emptyset$ is invertible and has itself asits inverse: $\emptyset \cup \emptyset = \emptyset \cup \emptyset = \emptyset$

Theorem: Let $(A, *)$ be a monoid. If $a \in A$ has an inverse, then that inverse is unique.

Proof: By contradiction: Assume not, then $\exists a \in A$ s.t. both b and c in A are its inverses, **i.e.** $a * b = b * a = e$, the identity element of $(A, *)$ and $a * c = c * a = e$ and $b \neq c$, then $b = b * e = b * (a * c) = (b * a) * c = e * c = c$.

qed

Since every invertible element of a for $(A, *)$ a monoid has a unique inverse, we can denote the inverse by the more standard notation a^{-1}.

Next, we need to understand inverses of elements obtained via the binary operation:

Theorem: Let $(A, *)$ be a monoid and let a, b be invertible elements of A. $a * b$ is also invertible and $a * b^{-1} = b^{-1} * a^{-1}$.

Remark: You might remember this formula from linear algebra when you looked at the inverse of a product of matrices AB.

Proof: Let e be the identity element of $(A, *) a * a^{-1} = a^{-1} * a = e$ and $b * b^{-1} = b^{-1} * b = e$. We would like to show $b^{-1} * a * -1$ is the inverse of $a * b$ by computing $(a * b) * (b^{-1} * a^{-1})$ and $(b^{-1} * a^{-1}) * (a * b)$ and showing both are e.

$(a * b) * (a^{-1} * b^{-1}) = a * (b * b^{-1}) * a^{-1} = a * e * a^{-1} = a * a^{-1} = e$

$(b^{-1} * a^{-1}) * (a * b) = b^{-1} * (a^{-1} * a) * b = b^{-1} * e * b = (b^{-1} * e) * b = b^{-1} * b = e$

Thus $b^{-1} * a^{-1}$ satisfies the conditions needed for it to be the inverse of $a * b$.Since an inverse of unique, $a * b$ is invertible and $b^{-1} * a^{-1}$.

Theorem: Let $(A, *)$ be a monoid, and let $a, b \in A$. Let $x \in A$ be invertible. $a = b * x \Leftrightarrow b = a * x^{-1}$. Similarly, $a = x * b \Leftrightarrow b = x^{-1} * a$

Proof: Let e be the identity element of $(A, *)$.

First $a = b * x \Leftrightarrow b = a * x^{-1}$:

$\Rightarrow$ Assume $a = b * x . a * x^{-1} = (b * x) * x^{-1} = b * x * x^{-1} = b * e = b$ as needed.

$\Leftarrow$ Assume $b = a * x^{-1}$. Then $b * x = (a * x^{-1}) = a * (x^{-1} * x) = a * e = 1$ as needed.

Apply the same type of argument to show $a = x * b \Leftrightarrow b = x^{-1} * a$.

qed

Let $(A, *)$ be a monoid. We can now make sense of a^n for $n \in \mathbb{Z}, n < -$ for every $n \in A$ invertible. Since n is a negative integer, $\exists p \in \mathbb{N}$ s.t. $n = -1$. Set $a^n = a^{-p} = (a^p)^{-1}$.

Theorem: Let $(A, *)$ be a monoid and let $a \in A$ be invertible. Then $a^n * a^m = a^{m+n} \qquad \forall m, n \in \mathbb{Z}$.

Proof: When $m \geq 0 \wedge n \geq n$ we have already proven this result. The rest of the proof splits into cases.

Case 1: $m = n \vee n = 0$

If $m = 0, n \in \mathbb{Z}, a^m * a^n = a^0 * a^n = e * a^n = a^n = a^{0+n}$ as needed.

If $m \in \mathbb{Z}, n = 0, a^m * a^n = a^m * a^0 = a^m * e = a^m = a^{m+0}$ as needed.

Case 2: $m < 0 \wedge n < 0$

$m < 0 \Rightarrow \exists p \in \mathbb{N}\ s.t.\ p = -m.\ n < 0 \Rightarrow \exists q \in \mathbb{N}\ s.t.\ q = -n.$

$a^m = a^{-p} = (a^p)^{-1} \wedge a^n = a^{-q} = (a^q)^{-1}$

$a^m * a^n = (a^p)^{-1} * (a^q)^{-1} = (a^q * a^p)^{-1} = (a^{p+q})^{-1} = a^{-(p+q)} = a^{-q-p} = a^{m+n} = a^{n+m}$

Case 3: $m \wedge n$ have opposite signs.

Without loss of generality, assume $m < 0 \wedge n > 0$ (the case $m > 0 \wedge n < 0$ is handled by the same argument). Since $m < 0, \exists p \in \mathbb{N}\ s.t.\ p = -m$. This case splits into two subcases:

Case 3.1: $m + n \geq 0$

Set $q = m + n$. Then $a^{m+n} = a^q = e * a^q = (a^p)^{-1} * a^p * a^q = (a^p)^{-1} * a^{p+q} = a^{-p} * a^{p+q} = a^m * a^{-m+m+n} = a^m * a^n$

Case 3.2: $m + n < 0$

Set $q = -(m+n) = -m-n \in \mathbb{N}^*$. Then $a^{m+n} = a^{-q} = (a^q)^{-1} * e = (a^q)^{-1} * (a^{-n} * a^n) = (a^q)^{-1} * (a^n)^{-1} * a^n = (a^n * a^q)^{-1} * a^n = (a^{n+p})^{-1} * a^n = (a^{n-m-n})^{-1} * a^n = (a^{-m})^{-1} * a^n = (a^p)^{-1} * a^n = a^m * a^n$

Theorem: Let $(A, *)$ be a monoid, and let a be an invertible element of A. $\forall m, n \in \mathbb{Z}, (a^m)^n = a^{mn}$.

Proof: We consider 3 cases:

Case 1: $n > 0$, **i.e.** $n \in \mathbb{N}^*$. $m \in \mathbb{Z}$ with no additional restrictions we proceed by induction on m.

Base Case: $n = 1 \qquad (a^m)^1 = a^m = a^{m\times 1}$

Inductive Step: We assume $(a^m)^n = a^{mn}$ and seek to prove $(a^m)^{n+1} = a^{m(n+1)}$. Start with $(a^m)^{n+1} = (a^m)^n * (a^m)^1 = a^{mn} * a^m = a^{mn+m} = a^{m(n+1)}$

Case 2: $n = 0$; no restriction on $m \in \mathbb{Z}$

$(a^m)^n = (a^m)^0 = e = a^0 = a^{m\times 0} = a^{mn}$

Case 3: $n < 0$; no restriction on $m \in \mathbb{Z}$.

Since $n < 0, \exists p \in \mathbb{N}\, s.t.\, p = -n$. By case 1, $(a^m)^p = a^{mp}$

$(a^m)^n = (a^m)^{-p} = ((a^m)^p)^{-1} = (a^{mp})^{-1} = a^{-mp} =^{m(-p)}= a^{mn}$

17.1 Groups

A motion formally defined in the 1870's even though theorems about groups proven as early as a century before that.

Definition: A group os a monoid in which every element is invertible. In other words, a group is a set A endowed with a binary operation $*$ satisfying the following properties:

1. $*$ is associative, **i.e.** $\forall x, y, z \in A, (x * y) * z = x * (y * z)$
2. There exists an identity element $e \in A$, **i.e.** $\exists e \in A\, s.t. \forall a \in A, a * e = e * a = a$
3. Every element of A in invertible, **i.e.** $\forall a \in A \exists a^{-1} \in A\ s.t.\ a * a^{-1} = a^{-1} * a = e$

Notation for Groups: $(A, *) \vee (\underbrace{A}_{set}, \underbrace{*}_{operation}, \underbrace{e}_{identity})$

Remark: Closure under the operation $*$ is <u>implicit</u> in the definition **i.e.** $\forall a, b \in A, a * b \in A$

Definition: A group $(A, *, e)$ is called <u>commutative</u> or <u>Abelian</u> if its operation $*$ is commutative.

Examples:

1. $(\mathbb{R}, +, 0)$ is an Abelian group.
   $-x$ is the inverse of $x, \forall x \in \mathbb{R}$
2. $(\mathbb{Q}^*, \times, 1) \qquad \mathbb{Q}^* = \mathbb{Q}^* \backslash \{0\} \qquad (\mathbb{Q}^*, \times, 1)$ is Abelian
   $\forall q \in \mathbb{Q}^*, q^{-1} = \frac{1}{q}$ is the inverse.
3. $(\mathbb{R}^3, +, 0)$ vectors in $\mathbb{R}^3$ with vector addition forms an Abelian group.
   $(x, y, z) + (x', y', z') = (x + x', y + y', z + z')$ vector addition.
   $0 = (0, 0, 0)$ is the identity. $\qquad (-x, -y, -z) = -(x, y, z)$ is the inverse of (x, y, z).

4. $(\tilde{\mathrm{M}}m, *, In)$ $n \times n$ invertible matrices with real coefficients under matrix multiplication with In as the identity elements forms a group which is NOT Abelian.

5. Set $A = \mathbb{Z}$ and recall the equivalence relation $x \equiv y \bmod 3$ **i.e.** $x \wedge y$ have te same remainder under the division by 3. Recall that $\mathbb{Z}/N = \{0, 1, 2\}$, **i.e.** the set of equivalence classes under the partition determined by this equivalence relation. We denote $\mathbb{Z}/N$ $\{0, 1, 2\} = \mathbb{Z}_3$

   Consider $(\mathbb{Z}_3, \oplus_3, 0)$ where $\oplus_3$ is the operation of addition modulo 3, **i.e.** $1 + 0 = 1, 1 + 1 = 2, 1 + 2 = 3 \equiv 0 \bmod 3$.

Claim: $(\mathbb{Z}_3, \oplus_3, 0)$ is an Abelian group.

Proof of Claim: Associativity of $\oplus_3$ follows from the associativity of $+$, addition of $\mathbb{Z}$. Clearly, 0 is the identity (don't forget 0 stands for all elements with remainder 0 under division by 3, **i.e.** $\{0, 3, -3, 6, -3, ...\}$). To compute inverses recall that $a \oplus_3 a^{-1} = 0, 0$ is the inverse of 0 because $0 + 0 = 0$. 2 is he inverse of 1 because $1 + 2 = 3 \equiv 0 \bmod 3$, and 1 is the inverse of 2 because $2 + 1 = 3 \equiv 0 \bmod 3$.

More generally, consider the equivalence relation on $\mathbb{Z}$ given by $x \equiv y \bmod n$ for $n \geq 1.\mathbb{Z}/N = \{0, 1, ..., n-1\} = \mathbb{Z}_n$. All possible remainders under division by n are the equivalence classes. Let $\oplus_n$ be addition mod n. By the same argument as above, $(\mathbb{Z}_n, \oplus_n, 0)$ is an Abelian group.

Q: What is we consider multiplication mod n, **i.e.** $\otimes_n$. Is $(\mathbb{Z}_n, \otimes_n, 1)$ a group?

A: No! $(\mathbb{Z}_n, \otimes_n, 1)$ is not even a monoid because $1 \otimes_n 0 = 0 \otimes_n 1 = 0$, so 1 is not an identity element for $\otimes_n$ on $\mathbb{Z}_n$.

Q: Can this be fixed?

A: Troubleshoot how to get rid of 0.

Consider $\mathbb{Z}_n^* = \mathbb{Z}_n \backslash \{0\} = \{1, 2, ..., n-1\}$ all non-zero elements in $\mathbb{Z}_n^*$. This eliminates 0 as an element, but can 0 arise any other way from the binary operation? It turns out the answer depends on n. If n is not prime, say $n = 6$, we get two divisors, **i.e.** elements that yield 0 when multiplied by precisely the factors of n, for $n = 6$, $\mathbb{Z}_6^* = \{1, 2, 3, 4, 5\}$ but $2 \otimes_6 3 = 6 \equiv 0 \bmod 6$, so $2 \wedge 3$ are two divisors.

Claim: If n is prime, then $(\mathbb{Z}_n^*, \otimes_n, 1)$ is an Abelian group.

Used in cryptology $\rightarrow$ you will see next semester.

As an example, let us look at the multiplication table for $\mathbb{Z}_5^*$ to see the inverse of various elements: $\mathbb{Z}_5^* = \mathbb{Z}_5 \backslash \{0\} = \{0, 1, 2, 3, 4, \} \backslash \{0\} - \{1, 2, 3, 4\}$

	1	2	3	4
1	1	2	3	4
2	2	4	1	2
3	3	1	4	2
4	4	3	2	1

$1^{-1} = 1 \quad 1 \otimes_5 1 = 1$
$2^{-1} = 3 \quad 2 \otimes_5 3 = 6 \equiv 1 \bmod 5$
$3^{-1} = 2 \quad 3 \otimes_5 2 = 6 \equiv 1 \bmod 5$
$4^{-1} = 1 \quad 4 \otimes_5 4 = 16 \equiv 1 \bmod 5$

The fact that $\mathbb{Z}_n^*, \otimes_n, 1$ is Abelian follows from the commutativity of multiplication on $\mathbb{Z}$.

6. Let $(A, *, e)$ be any group and let $a \in A$.
   Consider $A' = \{a^m \mid m \in \mathbb{Z}\}$ all powers of a. It turns out $(A', *, e)$ is a group called the cyclic group determined by a. $(A', *, e)$ is Abelian regardless of whether the original group was Abelian or not because of the theorem we proved on powers of a: $\forall m, n \in \mathbb{Z} \qquad a^m * a^n = a^{m+n} = a^{n+m} = a^n * a^m$.
   Cyclic groups come in two flavours: finite (A' is a finite set) and infinite (A' is an infinite set).
   For example, let $(A, *, e) = (\mathbb{Q}^*, , \times, 1)$
   If $a = -1 \qquad A' = \{(-1)^m \mid m \in \mathbb{Z}\} = \{-1, 1\}$ is finite.
   If $a = 2 \qquad A' = \{2^m \mid m \in \mathbb{Z}\} = \{1, 2, \frac{1}{2}, 4, \frac{1}{4}, ...\}$ is infinite.

18 Homomorphisms and Isomorphisms

Task: Understand the most natural functions between objects in abstract algebra such as semigroups, monoids or groups.

Definition: Let $(A, *)$ and $(B, *)$ ve vitg semigroups, monoids or groups. A function $f : A \to B$ is called a homomorphism if $f(x * y) = f(x) * f(y)\ \forall x, y \in A$. In other words, if f is a function that respects (behaves well with respect) to the binary operation.

Examples:

1. Consider $(\mathbb{Z}, +, 0)$ and $(\mathbb{R}^*, \times, 1)$.
   Pick $a \in \mathbb{R}^*$, then $f(n) = a^n$ is a homomorphism between $(\mathbb{Z}, +, 0)$ and $(\mathbb{R}^*, \times, 1)$ because $(\mathbb{R}^*, \times, 1)$ is a group, and we proved for groups that $a^{m+n} = f(m + n) = a^m * a^n = f(m) * f(n) \qquad \forall m, n \in \mathbb{Z}$.
2. More generally, $\forall a \in A$ invertible, where $(A, *)$ is a monoid with identity element e, $f(n) = a^m$ gives a homomorphism between $(\mathbb{Z}, +, 0)$ and $(A', *, e)*$, where as before $A' = \{a^m \mid m \in \mathbb{Z}\} \subset A$.
   We get even better behaviour if we require $f : A \to B$ to be bijective.

Definition: Let $(A, *)$ and $(B, *)$ both be semigroups, monoids or groups. A function $f : A \to B$ is called an isomorphism if $f : A \to B$ is both bijective AND a homomorphism.

Examples:

1. Let $A' = \{2^m \mid m \in \mathbb{Z}\} = \{1, 2, \frac{1}{2}, 4, \frac{1}{4}, ...\}$
   $f(m) = 2^m$ from $(\mathbb{Z}, +, 0)$ to $(A', \times, 1)$ is an isomorphism since $2^m \neq 2^n$ if $m \neq n$.
2. Let $A' = \{(-1)^m \mid m \in \mathbb{Z}\} = \{-1, 1\}$
   $f(m) = (-1)^m$ from $(\mathbb{Z}, +, 0)$ to $(A', \times, 1)$ is NOT an isomorphism since it's not injective $(-1)^2 = (-1)^4 = 1$.

Theorem: Let $(A, *)$ and $(B, *)$ both be semigroups, monoids or groups. The inverse $f^{-1} : B \to A$ of any isomorphism $f : A \to B$ from A to B is itself an isomorphism.

Proof: If $f : A \to B$ is an isomorphism $\Rightarrow f : A \to B$ is bijective $\Rightarrow f^{-1} : B \to A$ is bijective (proven when we discussed functions).

To show $f^{-1}B \to A$ is a homomorphism, let $u, v \in B$. $\exists x, y \in A$ s.t. $x = f^{-1}(u)$ and $y = f^{-1}(v)$, but then $u = f(x)$ and $v = f(y)$.

Since $f : A \to B$ is a homomorphism, $f(x * y) = f(x) * f(y) = u * v$. Then $f^{-1}(u * v) = f^{-1}(f(x * y)) = x * y = f^{-1}(u) * f^{-1}(v)$ as needed.

qed

Definition: Let $(A, *)$ and $(B, *)$ both be semigroups, monoids or groups. If $\exists f : A \to B$ an isomorphism betwen A and B, then $(A, *)$ and $(B, *)$ are said to be isomorphic.

Remark: "Ismorphic" comes from "iso" same + "morphé" form same abstract algebra structure on both $(A, *)$ and $(B, *)$ given to you in two different guises. As the French would say: "Même Marie, autre chapeau" same Mary, different hat.

19 Formal Languages

Task: Use what we learned about structues in abstract algebra in order to make sense of formal languages and grammars.
Let A be a finite set. When studying formal languages, we call A an alphabet and the elements of A letters.

Examples:

1. $A = \{0, 1\}$ binary digits
2. $A = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ decimal digits
3. $A =$ letters of the English alphabet

Definition: $\forall n \in \mathbb{N}^*$, we define a word of length n in the alphabet A as being any string of the form $a_1, a_2, ..., a_n \; s.t. \; a_i \in A \quad \forall i, 1 \leq i \leq n$. Let A^n be the set of all words of length n over the alphabet A.

Remark: There is a one-to-one correspondence between the string $a_1a_2...a_n$ and the ordered n-tuple $(a_1, a_2, ..., a_n) \in A^n = \underbrace{A \times ... \times A}_{n \; times}$ the Cartesian product of n copies of A.

Definition: Let $A^+ = \bigcup_{n=1}^{\infty} A^n = A^1 \cup A^2 \cup A^3 \cup ....$ A^+ is the set of all words of positive length over the alphabet A.

Examples:

1. $A = \{0, 1\}, A^+$ is the set of all binary strings of finite length that is at least on, **i.e.** $0, 1, 01, 10, 00, 11$, etc.
2. If $A =$ letters of the English alphabet, then A^+ consists of all non-empty strings of finite lengths of letters from the English alphabet. It is useful to also have the empty for ε in our set of strings. ε has length 0. Define $A^0 = \{\varepsilon\}$ and then adjoin the empty word ε to A^+. We get $A^* = \{\varepsilon\} \cup A^+ = A^0 \cup \overset{\infty}{\underset{n=1}{\cup}} A^n = \overset{\infty}{\underset{n=0}{\cup}} A^n$.

Notation: We denote the length of a word w by $\mid w \mid$. Next introduce an operation on A^*.

Definition: Let A be a finite set and let $w_1 \wedge w_2$ be words in A^*. $w_1 = a_1a_2...a_m \wedge w_2 = b_1b_2...b_n$. The concatenation of $w_1 \wedge w_2$ is the word $w_1 \circ w_2$, where $w_1 \circ w_2 = a_1a_2...a_mb_1b_2...b_n$. Sometimes $w_1 \circ w_2$ is denoted as just w_1w_2. Note that $\mid w_1 \circ w_2 \mid = \mid w_1 \mid + \mid w_2 \mid$.
Concatenation of words is:

1. associative
2. NOT commutative is A has more than one element.

Proof of (1): Let $w_1, w_2, w_3 \in A^*.w_1 = a_1a_2...a_m$ for some $m \in \mathbb{N}, w_2 = b_1b_2...b_n$ for some $m \in \mathbb{N}$ and $w_3 = c_1c_2...c_p$ for some $p \in \mathbb{N}$. $w_1 \circ w_2) \circ w_3 = w_1 \circ (w_2 \circ w_3) = a_1a_2...a_mb_1b_2...b_nc_1c_2...c_p$.

qed

Proof of (2): Since A has at least two elements, $\exists a, b \in A$ s.t. $a \neq b$.

$a \circ b = ab \neq ba = b \circ a$.

qed

A^* is closed under the operation of concatenation $\Rightarrow$ concatenation is a binary operation of A^* as $\forall w_1, w_2 \in \mathring{A}^*, w_1 \circ w_2 \in A^*$.

Theorem Let A be a finite set. $(A^*, \circ)$ is a monoid with identity element ε.

Proof: Concatenation $\circ$ is an associative binary operation on A^* as we showed above. Moreover, $\forall w \in A^*, \varepsilon \circ w = w \circ \varepsilon = w$, so ε is the identity element of A^*.

qed

Definition: Let A be a finite set. A language over A is a subset of A^*. A language L over A is called a formal language is $\exists$ a finite set of rules of algorithm that generates exactly L, **i.e.** all words that belong to L and no other words.

Theorem: Let A be a finite set.

1. If L_1 and L_2 are languages over $A, L_1 \cup L_2$ is a language over A.
2. If L_1 and L_2 are languages over $A, L_1 \cap L_2$ is a language over A.
3. If L_1 and L_2 are languages over A, the concatenation of $L_1 \wedge L_2$ given by $L_1 \circ L_2 = \{w_1 \circ w_2 \in A^* \mid w_1 \in L_1 \wedge w_2 \in L_2\}$ is a language over A.
4. Let L be a language over A. Define $L^1 = L$ inductively for and $n \geq 1 \quad L^n = L \circ L^{n-1}.L^n$ is a language over A. Furthermore, $L^* = \{\varepsilon\} \cup L^! \cup L^2 \cup L^3 \cup ... = \overset{\infty}{\underset{n=0}{\cup}} L^n$ is a language over A.

Proof: By definition, a language over A is a subset of A^*. Therefore, if $L_1 \subseteq A^* \wedge L_2 \subseteq A^*$, then $L_1 \cup L_2 \subseteq A^* \wedge L_1 \cap L_2 \subseteq A^*.\forall w_1 \circ w_2 \in L_1 \circ L_2, w_1 \circ w_2 \in A^*$ becomes $w_1 \in A^n$ for some n and $w_2 \in A^m$ for some m so $w_1 \circ w_2 \in A^{m+n} \subseteq A^* = \overset{\infty}{\underset{n=1}{\cup}} A^n$.

Applying the same reasoning inductively, we see that $L \subset A^* \Rightarrow L^* \subseteq A^*$ as $L^m \subseteq A^* \, \forall n \geq 0$.

qed

Remark: This theorem gives us a theoretic way of building languages, but we need a practical way. The practical way of building a language is through the notion of a grammar.

Definition: A (formal) grammar is a set of production rules for strings in a language.
To generate a language we use:

1. A the set, which is the alphabet of the language.
2. A start symbol <s>
3. A set of production rules.

Example: $A = \{0, 1\}$; start symbol <s>; 2 production rules given by:

1. <s> $\rightarrow$ 0<s>1
2. <s> $\rightarrow$ 01

Let's see what we generate: via rule 2 <s> $\rightarrow$ 01, so we get <s> $\Rightarrow$ 01
Via rule 1 <s> $\rightarrow$ 0<s>1, then via rule 2, 0<s>1 $\rightarrow$ 0011. We write the process as <s> $\rightarrow$ 0<s>1 $\Rightarrow$ 0011.
Via rule 1, <s> $\rightarrow$ 0<s>1, then via rule 1 again 0<s>1 $\rightarrow$ 00<s>11, then via rule 2, 00<s>11 $\rightarrow$ 000111.
We got <s> $\Rightarrow$ 0<s>1 $\Rightarrow$ 00<s>11 $\Rightarrow$ 000111.
The language L we generated thus consists of all strings of the form $0^m 1^m$ (m 0's followed by m 1's) for all $m \geq 1, m \in \mathbb{N}$
We saw 2 types of strings that appeared in this process of generating L:

1. terminals, **i.e.** the elements of A
2. nonterminals, **i.e.** strings that don't consist solely of 0's and 1's such as <s>, 0<s>1, 00<s>11, etc.

The production rules then have the form:

nonterminal $\rightarrow$ word over the alphabet V = terminals, non-terminals

<T> $\rightarrow$ w

In our notation, the set of nonterminals if $V\backslash A$, so <T>$\in V\backslash A \wedge w \in V^* - \overset{\infty}{\underset{n=0}{\cup}} V^n$. To the production rule <T>$\rightarrow w$.
We can associate the ordered pair $(<\text{T}>, w) \in (V\backslash A) \times V^*$, so the set of production rules, which we will denote by P, is a subset of the Cartesian product $(V\backslash A) \times V^*$.
Grammers come in two flavours:

1. Contest - free grammars where we can replace any occurrence of <T> by w if <T>$\rightarrow w$ is one of our production rules.
2. Context - sensitive grammars only certain replacements of <T> by w are allowed, which are governed by the syntax of our language L.

The example we have was of a context free grammar. We can now finally define context free grammars.

Definition: A context free grammar $(V, A, <\text{s}>\ P)$ consists o a finite set V, a subset A of V, an element <s> of $V\backslash A$, and a finite subset P of the Cartesian product $V\backslash A \times V^*$.

Notation: $(\underset{set\ of\ terminals\ and\ non\ terminals}{V}, \underset{set\ of\ terminals}{A}, \underset{startsymbol}{<s>}, \underset{set\ of\ production\ rules}{P})$

Example: $A = \{0, 1\}$; start symbol <s>; 3 production rules given by:

1. <s>$\rightarrow$ 0<s>1
2. <s>$\rightarrow$ 01
3. <s>$\rightarrow$ 0011

We notice here that the word 0011 can be generated in 2 ways in this context free grammar:

By rule 3, <s>$\rightarrow$ 0011 so <s>$\Rightarrow$ 0011

$\vee$

By rule 1, <s>$\rightarrow$ 0<s>1 and by rule 2, 0<s>1 $\rightarrow$ 0011. Therefore, <s>$\Rightarrow$ 0<s>1 $\Rightarrow$ 0011.

Definition: A grammar is called ambiguous if it generates the same string in more than one way.
Obviously, we prefer to have unambiguous grammars, else we waste computer operations.
Next, we need to spell out how words relate to each other in the production of our language via the grammar:

Definition: Let w' and w" be words over the alphabet V = terminals, nonterminals. We say that w' directly yields w" if $\exists$ words $u \wedge v$ over the alphabet V and a production rule <T>$\rightarrow w$ of the grammar s.t. $w' = u$ <T> $\wedge w$" $= uwv$, where either or both of the words u and v may be the empty word.
In other words, w' directly yields w" $\Leftrightarrow \exists$ production rule <T>$\rightarrow w$ in the grammar s.t. w" may be obtained from w' by replacing a simple occurrence of the nonterminal <T> within the word w' by the word w.

Notation: w' directly yields w" is denoted by $w' \Rightarrow w$"

Definition: Let $w' \wedge w$" be words over the alphabet V. We say that w' yields w" if either $w' = w$" or else $\exists$ words $w_0, w_1, ...w_n$ over the alphabet V s.t. $w_0 = w', w_n = w", w_{i-1} \Rightarrow w_i$ for all $i, 1 \leq i \leq n$. In other words, $w_0 \Rightarrow w_1 \Rightarrow w_2 \Rightarrow ... \Rightarrow w_n - 1 \Rightarrow w_n$

Notation: w' yields w" is denotes by $w' \stackrel{*}{\Rightarrow} w$".

Definition: Let $(V, A, <s>, P)$ be a context free grammar. The language generated by this grammar is the subset L or A^* defined by $L = \{w \in A^* \mid <s> \stackrel{*}{\Rightarrow} w\}$
In other words, th language L generated by a context free grammar $(V, A, <s>, P)$ consists of the set of all finite strings consisting entirely of terminals that may be obtained from the start symbol <s> by applying a finite sequence of production rules of the grammar where the application of one production rule causes one and only one nonterminal to be replaced by the string in V^* corresponding of the right hand side of the production rule.

19.1 Phrase Structure Grammars

Definition: A phrase structure grammar $(V, A <s>, P)$ consists of a finite set V, a subset A of V, an element <s> of $V \backslash A$, and a finite subset P of $(V^* \backslash A^*) \times V^*$
In a context free grammar, the set of production rules $P \subset (V \backslash A) \times V^*$.
In a phrase structure grammar, $P \subset (V^* \backslash A^*) \times V^*$. In other words, a production rule in a phrase structure grammar $r \rightarrow w$ has a left hand side n that may contain more than one nonterminal. It is required to contain at least one nonterminal.
For example, if $A\{0, 1\}$ and <s> is the start symbol in a phrase grammar

grammar, 0<s>0<s>0 $\rightarrow$ 00010 would be an acceptable production rule in a phrase structure grammar but not in a context free grammar.
The notions $w' \Rightarrow w$" (w' directly yields w") and $w' \overset{*}{\Rightarrow} w$" ($w'$ yields w") are defined the same way as for context free grammars except that our production rules may, of course, be more general as we saw in the example above.

Definition: Lt $(V, A <s>, P)$ be a phrase structure grammar. The language generated by this grammar is the subset L or A^* defined by $L\{w \in A^* |<s>\overset{*}{\Rightarrow} w\}$

Remark: The term phrase structure grammars was introduced by Noam Chowsky.

Definition: A lamguage L generated by a context-free grammar is called a context-free language.

We now want to understand a particularly important subclass of context free languages called regular languages.

20 Regular Languages

Task: Understand when a language is regular and how regular languages are produced. Understand basics of automata theory.

History: The term regular languages was introduced by Stephen Kleene in 1951. A more descriptive name is finite-state languages as we will see that a language is regular $\Leftrightarrow$ it can be recognised by a finite state acceptor, which is a type of finite state machine.
The definition of a regular language is very abstract, though. First, describe what operations the collection of regular languages is closed under: Let A be a finite set, and let $A*$ be the set of all words over the alphabet A. The regular language over the alphabet A consitute the smallest collection C of subsets of $A*$ satisfying that:

1. All finite subsets of $A*$ belong to C.
2. C is closed under the Kleene start operation (if $M \subseteq A*$ is inside C, **i.e.**$M \subseteq C$, then $M* \subseteq C$)
3. C is closed under concatenation (if $M \subseteq A*, N \subseteq A*$ satisfy that $M \subseteq C \wedge N \subseteq C$, then $M \circ N \subseteq C$)
4. C is closed under union (if $M \subseteq A* \wedge N \subseteq A*$ satisfy that $M \subseteq C \wedge N \subseteq C$, then $M \cup N \subseteq C$)

Definition: Let A be a finite set, and let $A*$ be the set of words over the alphabet A. A subset L of $A*$ is called a regular language over the alphabet A is $L = L_m$ for some finite sequence $L_1, L_2, ..., L_m$ os subsets of $A*$ with the property that $\forall i, 1 \leq i \leq m, L_i$ satisfies one of the following:

1. L_i is a finite set
2. $L_i = L_j*$ for some $j, 1 \leq j \leq i$ (the Klenne star operation applied to one of the previous $L'_j s$)
3. $L_i = L_j \circ L_k$ for some j, k such that $1 \leq j, k < i$ (L_i is a concatenation of previous $L'_j s$)
4. $L_i = L_j \cup L_k$ for some j, k such that $1 \leq k, j < i$ (L_i is a union of previous $L'_j s$)

Example 1: Let $A = \{0, 1\}$. Let $L = \{0^M 1^n \mid m, n \in \mathbb{N} \quad m \geq 0, n \geq 0\}$
L is a regular language. Note that L consists of all strings of first 0's, then 1's or the empty string ε.$0^m 1^n$ stands for m 0's followed by n 1's, **i.e.** $0^m \circ 1^n$. Let us examine $L' = \{0^m \mid m \in \mathbb{N}, m \geq 0\}$ and $L" = \{1^n \mid n \in \mathbb{N}, n \geq 0\}$

Q: Can we obtain them via operatons listed among 1-4?

A: Yes! Let $M = \{0\}$ $M \subseteq A \subseteq A*$ and $M* = L^1 = \{0^m \mid m \in \mathbb{N} \quad m \geq 0\}$. Let $N = \{1\}$ $N \subseteq A \subseteq A*$ and $N* = L" = \{1^n \mid n \in \mathbb{N}, n \geq 0\}$. In other words, we can do $L_1 = \{0\}, L_2 = \{1\}, L_3 = L_1*, l_4 = L_2*, L_5 = L_4 \circ L_5 = L$. Therefore, L is a regular language.

Example 2 Let $A = \{0, 1\}$. Let $L = \{0^m 1^m \mid m \in \mathbb{N}, m \geq 1\}$. L is the language we used as an eample earlier. It turns out L is NOT regular. This language consists of strings of 0's followed by an equal number of strings of 1's. For a machine to decide that the string $0^m 1^m$ is inside the language is must store the number of 1's, as it examines the number of 0's or vice versa. The number of strings of the type $0^m 1^m$ is not finite, however, so a finite-state machine cannot recognise this language. Heuristically, regular languages correspond to problems that can be solved with finite memory, **i.e.** we only need to remember one of finitely many things. By contrast, nonregular languages correspond to problems that cannot be solved with finite memory.

Theorem: The collection of regular languages L is also closed under the following two operations:

1. Intersection, **i.e.** if $L', L"$ are regular languages (**i.e.** $L' \cup L" \in C$) then their intersection $L' \cap L"$ is a regular language.
2. Complement, **i.e.** if L is a regular language (**i.e.** $L \in C$), then $A* \backslash L$ is a regular language ($A* \backslash L \in C$).

Remark: These two properties did not come into the definition of a regular language, but they are true and often quite useful.

20.1 Finite State Acceptors and Automata Theory

Definition: An automation is a mathematical model of a computing device. Plural of automation is automata.

Basic idea: Reason about compatability without having to worry about the complexity of actual implementation.
It is most reasonable to consider at the beginning just finite states automata, **i.e.** machines with a finite number of internal states. The data entered discretely, and each datum causes the machine to either remain in the same internal state or else make the transition to some other state determined solely by 2 pieces of information:

1. The current state
2. The input datum

In other words, if S is the finite set of all possible states of our finite state machine, then the transition mapping t that tells us how the internal state of the machine changes on inputting a datum will depend on the current state $s \in S$ and the imput datum a, **i.e.** the machine will enter a (potentially) new state $s' = t(s, a)$.

Want to use finite state machines to recognise languages over some alphabet A. Let L be our language.
Since our finite state machine accepts (**i.e.** returns yes to) w if $w \in L$,

$$\begin{array}{cc} \underline{\text{Input}} & \underline{\text{Output}} \\ \text{Word } w = a_1...a_n, a_i \in A \forall i & \begin{array}{c}\text{Yes if } w \in L \\ \text{No if } w \ni L\end{array} \end{array}$$

we call our machine a finite state acceptor. We want to give a rigorous definition of a finite state acceptor. To check $w = a_1...a_n$, we input each a_i starting with a_1 and trace how the internal state of the machin changes. S is our set of states of the machine (a finite set). The transition mapping t takes the pair(s, a) and returns the new state $s' = t(s, a)$ (where $s \in S \wedge a \in A$) that the machine has reached so $t : S \times A \to S$.
Some elements and subets of S are important to understand:

1. The initial state $i]inS$ where the machine starts
2. The subset $F \subseteq S$ of finishing states

It turns out that knowing S, F, i, t, W satisfies a finite state acceptor completely.

Definition: A finite state acceptor (S, A, i, t, F) consists of a finite set S of states, a finite set A that is the input alphabet, a starting state $i \in S$, a transition mapping $t : S \times A \to S$, and a set F of finishing states, where $F \subseteq S$.

Definition: Let (S, A, i, tF) be a finite state acceptor, and let $A*$ denote the set of words over the input alphabet A. A word $a_1, a_2...a_n$ of length n over the alphabet A is said to be recognised or accepted by the finite state acceptor if $\exists s_0, s_1, ..., s_n \in S$ states s.t. $s_0 = i$ (the initial state), $s_n \in F$, and $s_i = t(s_{i-1}, a_i) \forall i \quad 1 \leq i \leq n$.

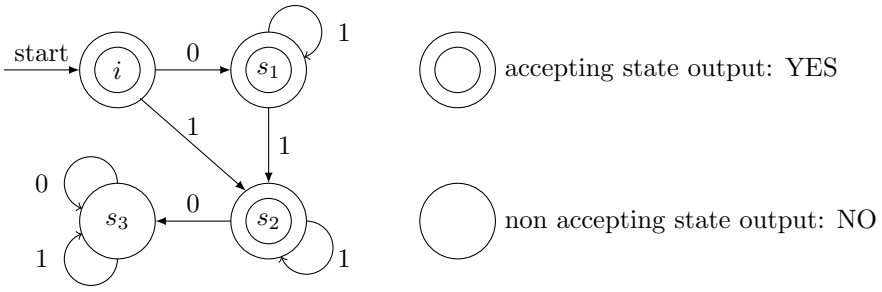
Definition: Let (S, A, i, t, F) be a finite state acceptor. A language L over the alphabet A is said to be recognised or accepted by the finite state acceptor. In the definition of a finite state acceptor, t is the transition maping, which may or may not be a function (hence the careful terminology). This is because finite state acceptors come in 2 flavours:

1. Deterministic: every state has exactly one transition for each possible input, **i.e.** $\forall (s, a) \in S \times A \ \exists!\ t(s, a) \in S$. In other words, the transition mapping is a function.
2. Non-deterministic: an input can lead to one, more than one or no transition for a given state. Some $(s, a) \in S \times A$ might be assigned to more than one element of S, **i.e.** the transition mapping is not a function.

Surprisingly $\exists$ algorithm that transforms a non deterministic (thought more complex one) using the powerset construction.
As a result, we have the following theorem:

Theorem: A language L over som alphabet A is a regular language $\Leftrightarrow$ L is recognised by a deterministic finite state acceptor with input alphabet $A \Leftrightarrow L$ is recognised by a nondeterministic finite state acceptor with input alphabet A.

Example: Build a deterministic finite state acceptor for the regular language $L = \{0^m 1^n \mid m, n \in \mathbb{N}, m \geq 0, n \geq 0\}$



Accepting states in this examples: i, s_1, s_2
Non accepting states: s_3
Start states: i