1 Mathematical Preliminaries and Notation

Sets

A **set** is a collection of elements, without any structure other than membership.
The usual set operations are **union** ($\cup$), **intersection** ($\cap$), **difference** ($-$) and **complementation** defined as

$$\begin{aligned} S_1 \cup S_2 &= \{\, x : x \in S_1 \text{ or } x \in S_2 \,\}, \\ S_1 \cap S_2 &= \{\, x : x \in S_1 \text{ and } x \in S_2 \,\}, \\ S_1 - S_2 &= \{\, x : x \in S_1 \text{ and } x \notin S_2 \,\}, \\ \overline{S} &= \{\, x : x \in U \text{ and } x \notin S \,\}. \end{aligned}$$

DeMorgan's laws

$$\begin{aligned} \overline{S_1 \cup S_2} &= \overline{S_1} \cap \overline{S_2}, \\ \overline{S_1 \cap S_2} &= \overline{S_1} \cup \overline{S_2}. \end{aligned}$$

A set S_1 is said to be a **subset** of S if every element of S_1 is also an element of S. We write this as

$$S_1 \subseteq S.$$

If $S_1 \subseteq S$, but S contains an element not in S_1, we say that S_1 is a **proper subset** of S; we write this as

$$S_1 \subset S.$$

If S_1 and S_2 have no common element, then the sets are said to be **disjoint**. We write this as

$$S_1 \cap S_2 = \varnothing.$$

A set is said to be finite if it contains a **finite** number of elements; otherwise it is **infinite**.
The set of all subsets of a set S is called the **powerset** of S and is denoted by 2^S. If S is finite, then

$$|2^S| = 2^{|S|}.$$

The sets whose elements are ordered sequences of elements from other sets are said to be the **Cartesian product** of other sets. For the Cartesian product of n sets, which itself is a set of ordered pairs, we write

$$S = S_1 \times S_2 \times \cdots \times S_n = \{\, (x_1,\ x_2,\ \cdots,\ x_n) : x_i \in S_i \,\}.$$

Suppose that $S_1,\ S_2,\ \cdots,\ S_n$ are subsets of a given set S and that the following holds:

1. The subsets $S_1,\ S_2,\ \cdots,\ S_n$ are mutually disjoint;
2. $S_1 \cup S_2 \cup \cdots \cup S_n = S$;
3. none of the S_i is empty.

Then $S_1,\ S_2,\ \cdots,\ S_n$ is called a **partition** of S.

Functions and Relations

A **function** is a rule that assigns to elements of one set a unique element of another set. If f denotes a function, then the first set is called the **domain** of f, and the second set is its **range**. We write

$$f : S_1 \to S_2$$

to indicate that the domain of f is a subset of S_1 and that the range of f is a subset of S_2. If the domain of f is all of S_1, we say that f is a **total function** on S_1; otherwise f is said to be a **partial function**.

Let $f(n)$ and $g(n)$ be functions whose domain is a subset of the positive integers. We say that

1. f has **order at most** g if there exists a positive constant c such that for all sufficiently large n
$$f(n) \leqslant c|g(n)| \qquad \xrightarrow{\text{expressed as}} \qquad f(n) = O(g(n)).$$
2. f has **order at least** g if there exists a positive constant c such that for all sufficiently large n
$$f(n) \geqslant c|g(n)| \qquad \xrightarrow{\text{expressed as}} \qquad f(n) = \Omega(g(n)).$$
3. f and g have the **same order of magnitude** if there exist constant c_1 and c_2 such that for all sufficiently large n
$$c_1|g(n)| \leqslant |f(n)| \leqslant c_2|g(n)| \qquad \xrightarrow{\text{expressed as}} \qquad f(n) = \Theta(g(n)).$$

Some functions can be represented by a set of pairs

$$\{\ (x_1, y_1), (x_2, y_2), \cdots \}.$$

where x_i is an element in the domain of the function, and y_i is the corresponding value in its range. For such a set to define a function, each x_i can occur at most once as the first element of a pair. If this is not satisfied, the set is called a **relation**.

Equivalence is a generalization of the concept of equality (identity). A relation denoted by $\equiv$ is considered an equivalence if it satisfies three rules:

1. The reflexivity rule
$$x \equiv x \text{ for all } x;$$
2. The symmetry rule
$$\text{if } x \equiv y, \text{ then } y \equiv x;$$
3. The transitivity rule
$$\text{if } x \equiv y \text{ and } y \equiv z, \text{ then } x \equiv z.$$

If S is a set on which we have a defined equivalence relation, then we can use this equivalence to partition the set into **equivalence classes**.