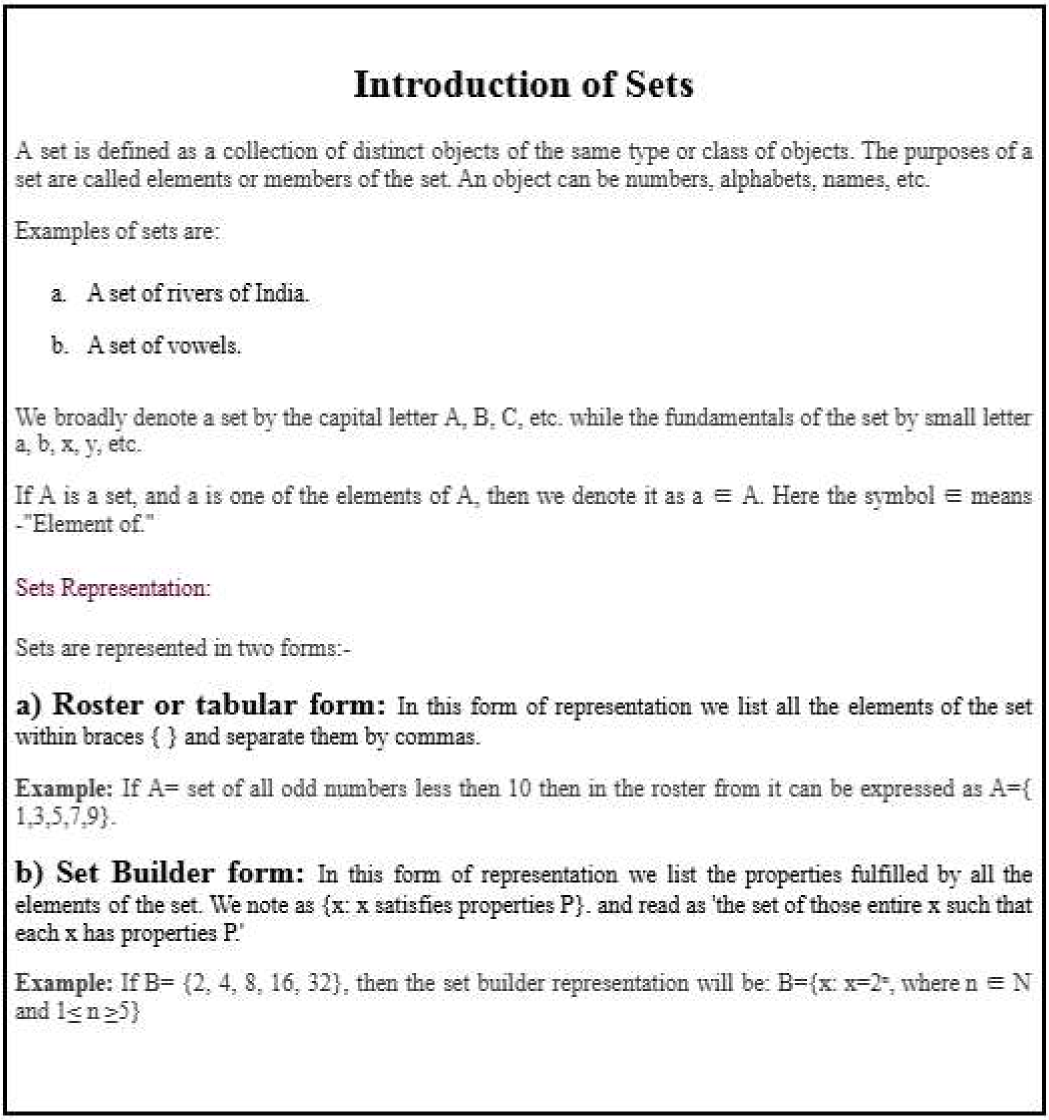


Introduction of Sets

A set is defined as a collection of distinct objects of the same type or class of objects. The purposes of a set are called elements or members of the set. An object can be numbers, alphabets, names, etc.

Examples of sets are:

a. A set of rivers of India.

b. A set of vowels.

We broadly denote a set by the capital letter A, B, C, etc. while the fundamentals of the set by small letter a, b, x, y, etc.

If A is a set, and a is one of the elements of A, then we denote it as a ≡ A. Here the symbol ≡ means -"Element of."

Sets Representation:

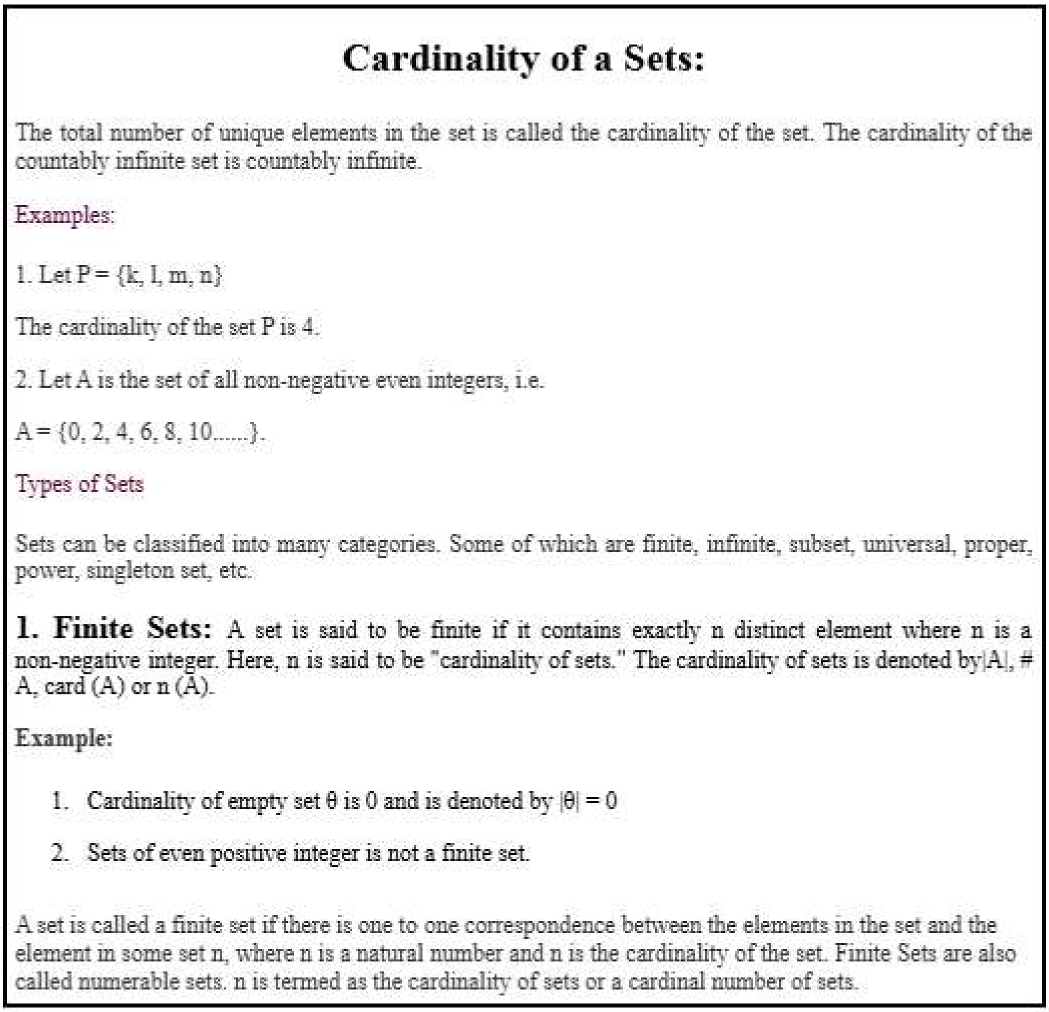
Sets are represented in two forms:-

a) Roster or tabular form: In this form of representation we list all the elements of the set within braces { } and separate them by commas.

Example: If A= set of all odd numbers less then 10 then in the roster from it can be expressed as A={ 1,3,5,7,9}.

b) Set Builder form: In this form of representation we list the properties fulfilled by all the elements of the set. We note as {x: x satisfies properties P}. and read as 'the set of those entire x such that each x has properties P.'

Example: If B= {2, 4, 8, 16, 32}, then the set builder representation will be: B={x: $x=2^n$, where n ≡ N and 1≤ n ≥5}



Cardinality of a Sets:

The total number of unique elements in the set is called the cardinality of the set. The cardinality of the countably infinite set is countably infinite.

Examples:

1. Let P = {k, l, m, n}

The cardinality of the set P is 4.

2. Let A is the set of all non-negative even integers, i.e.

A = {0, 2, 4, 6, 8, 10......}.

Types of Sets

Sets can be classified into many categories. Some of which are finite, infinite, subset, universal, proper, power, singleton set, etc.

1. Finite Sets: A set is said to be finite if it contains exactly n distinct element where n is a non-negative integer. Here, n is said to be "cardinality of sets." The cardinality of sets is denoted by|A|, # A, card (A) or n (A).

Example:

1. Cardinality of empty set θ is 0 and is denoted by |θ| = 0
2. Sets of even positive integer is not a finite set.

A set is called a finite set if there is one to one correspondence between the elements in the set and the element in some set n, where n is a natural number and n is the cardinality of the set. Finite Sets are also called numerable sets. n is termed as the cardinality of sets or a cardinal number of sets.

2. Infinite Sets: A set which is not finite is called Infinite Sets.

3. Countably Infinite: If there is one to one correspondence between the elements in set and element in N. A countably infinite set is also known as Denumerable. A set that is either finite or denumerable is known as countable. A set which is not countable is known as Uncountable. The set of a non-negative even integer is countable Infinite.

Uncountable Infinite: A set which is not countable is called Uncountable Infinite Set or non-denumerable set or simply Uncountable.

Example: Set R of all +ve real numbers less than 1 that can be represented by the decimal form 0. a_1, a_2, a_3..... Where a_i is an integer such that $0 \leq a_i \leq 9$.

4. Subsets: If every element in a set A is also an element of a set B, then A is called a subset of B. It can be denoted as $A \subseteq B$. Here B is called Superset of A.

Example: If A= {1, 2} and B= {4, 2, 1} the A is the subset of B or $A \subseteq B$.

Properties of Subsets:

1. Every set is a subset of itself.
2. The Null Set i.e.$\emptyset$ is a subset of every set.
3. If A is a subset of B and B is a subset of C, then A will be the subset of C. If $A \subset B$ and $B \subset C \Rightarrow A \subset C$
4. A finite set having n elements has 2^n subsets.

5. Proper Subset: If A is a subset of B and $A \neq B$ then A is said to be a proper subset of B. If A is a proper subset of B then B is not a subset of A, i.e., there is at least one element in B which is not in A.

Example:

(i) Let A = {2, 3, 4}

B = {2, 3, 4, 5}

A is a proper subset of B.

(ii) The null ∅ is a proper subset of every set.

6. Improper Subset: If A is a subset of B and A = B, then A is said to be an improper subset of B.

Example

(i) A = {2, 3, 4}, B = {2, 3, 4}

A is an improper subset of B.

(ii) Every set is an improper subset of itself.

7. Universal Set: If all the sets under investigations are subsets of a fixed set U, then the set U is called Universal Set.

Example: In the human population studies the universal set consists of all the people in the world.

8. Null Set or Empty Set: A set having no elements is called a Null set or void set. It is denoted by∅.

9. Singleton Set: It contains only one element. It is denoted by {s}.

Example: $S= \{x|x \in N, 7<x<9\} = \{8\}$

10. Equal Sets: Two sets A and B are said to be equal and written as A = B if both have the same elements. Therefore, every element which belongs to A is also an element of the set B and every element which belongs to the set B is also an element of the set A.

1. $A = B \Leftrightarrow \{x \in A \Leftrightarrow x \in B\}$.

If there is some element in set A that does not belong to set B or vice versa then $A \neq B$, i.e., A is not equal to B.

11. Equivalent Sets: If the cardinalities of two sets are equal, they are called equivalent sets.

Example: If A= {1, 2, 6} and B= {16, 17, 22}, they are equivalent as cardinality of A is equal to the cardinality of B. i.e. $|A|=|B|=3$

12. Disjoint Sets: Two sets A and B are said to be disjoint if no element of A is in B and no element of B is in A.

Example:

R = {a, b, c}

S = {k, p, m}

R and S are disjoint sets.

13. Power Sets: The power of any given set A is the set of all subsets of A and is denoted by **P (A)**. If A has n elements, then **P (A)** has $\mathbf{2^n}$ elements.

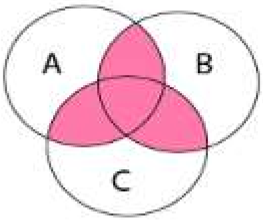
Example: A = {1, 2, 3}

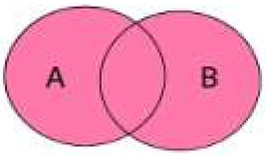
P (A) = {ø, {1}, {2}, {3}, {1, 2}, {1, 3}, {2, 3}, {1, 2, 3}}.

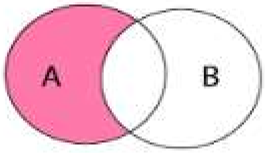
Venn Diagrams:

Venn diagram is a pictorial representation of sets in which an enclosed area in the plane represents sets.

Examples:







Operations on Sets

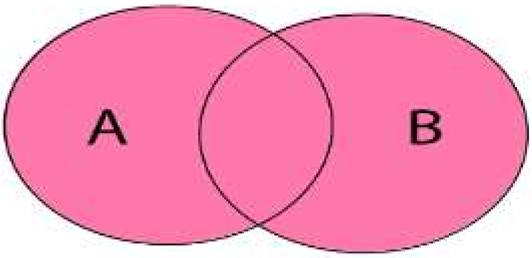
The basic set operations are:

1. Union of Sets: Union of Sets A and B is defined to be the set of all those elements which belong to A or B or both and is denoted by A∪B.

1. $A \cup B = \{x: x \equiv A \text{ or } x \equiv B\}$

Example: Let A = {1, 2, 3}, B= {3, 4, 5, 6}

A∪B = {1, 2, 3, 4, 5, 6}.

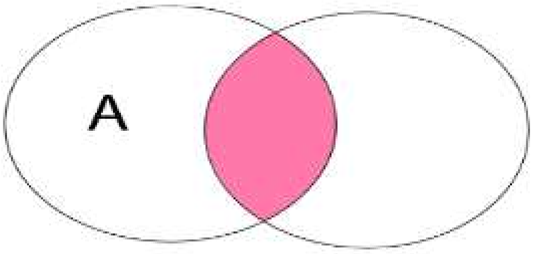


2. Intersection of Sets: Intersection of two sets A and B is the set of all those elements which belong to both A and B and is denoted by A ∩ B.

1. $A \cap B = \{x: x \equiv A \text{ and } x \equiv B\}$

Example: Let A = {11, 12, 13}, B = {13, 14, 15}

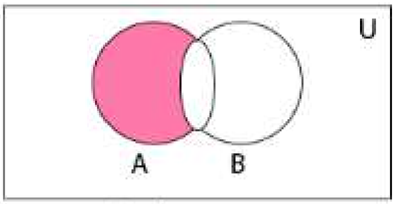
A ∩ B = {13}.



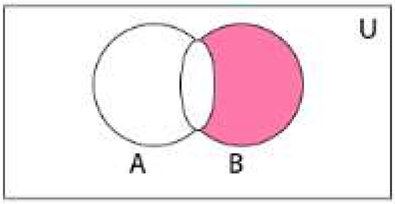
3. Difference of Sets: The difference of two sets A and B is a set of all those elements which belongs to A but do not belong to B and is denoted by A - B.

1. $A - B = \{x: x \in A \text{ and } x \notin B\}$

Example: Let $A = \{1, 2, 3, 4\}$ and $B = \{3, 4, 5, 6\}$ then $A - B = \{3, 4\}$ and $B - A = \{5, 6\}$



A - B



B - A

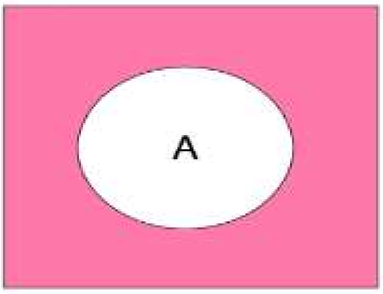
4. Complement of a Set: The Complement of a Set A is a set of all those elements of the universal set which do not belong to A and is denoted by A^c.

$A^c = U - A = \{x: x \in U \text{ and } x \notin A\} = \{x: x \notin A\}$

Example: Let U is the set of all natural numbers.

$A = \{1, 2, 3\}$

A^c = {all natural numbers except 1, 2, and 3}.



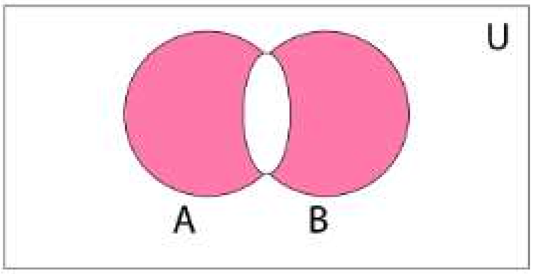
5. Symmetric Difference of Sets: The symmetric difference of two sets A and B is the set containing all the elements that are in A or B but not in both and is denoted by A ⊕ B i.e.

1. $A \oplus B = (A \cup B) - (A \cap B)$

Example: Let A = {a, b, c, d}

B = {a, b, l, m}

A ⊕ B = {c, d, l, m}

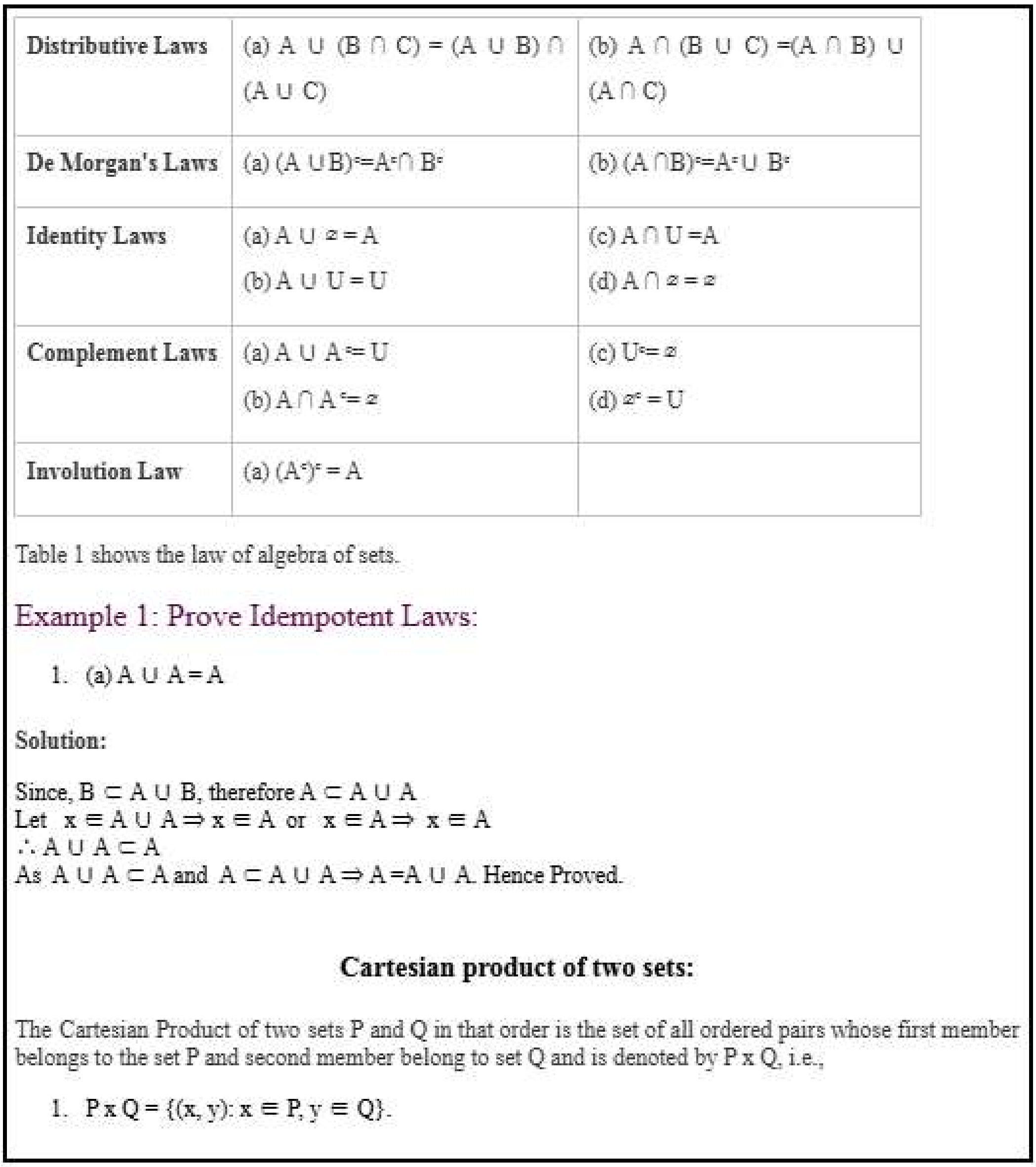


Algebra of Sets

Sets under the operations of union, intersection, and complement satisfy various laws (identities) which are listed in Table 1.

Table: Law of Algebra of Sets

Idempotent Laws	(a) $A \cup A = A$	(b) $A \cap A = A$
Associative Laws	(a) $(A \cup B) \cup C = A \cup (B \cup C)$	(b) $(A \cap B) \cap C = A \cap (B \cap C)$
Commutative Laws	(a) $A \cup B = B \cup A$	(b) $A \cap B = B \cap A$



Distributive Laws	(a) $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$	(b) $A \cap (B \cup C) =(A \cap B) \cup (A \cap C)$
De Morgan's Laws	(a) $(A \cup B)^c=A^c\cap B^c$	(b) $(A \cap B)^c=A^c\cup B^c$
Identity Laws	(a) $A \cup \varnothing = A$ (b) $A \cup U = U$	(c) $A \cap U =A$ (d) $A \cap \varnothing = \varnothing$
Complement Laws	(a) $A \cup A^c= U$ (b) $A \cap A^c = \varnothing$	(c) $U^c= \varnothing$ (d) $\varnothing^c = U$
Involution Law	(a) $(A^c)^c = A$	

Table 1 shows the law of algebra of sets.

Example 1: Prove Idempotent Laws:

1. (a) $A \cup A = A$

Solution:

Since, $B \subset A \cup B$, therefore $A \subset A \cup A$
Let $x \equiv A \cup A \Rightarrow x \equiv A$ or $x \equiv A \Rightarrow x \equiv A$
$\therefore A \cup A \subset A$
As $A \cup A \subset A$ and $A \subset A \cup A \Rightarrow A =A \cup A$. Hence Proved.

Cartesian product of two sets:

The Cartesian Product of two sets P and Q in that order is the set of all ordered pairs whose first member belongs to the set P and second member belong to set Q and is denoted by P x Q, i.e.,

1. $P \times Q = \{(x, y): x \equiv P, y \equiv Q\}$.

Example: Let P = {a, b, c} and Q = {k, l, m, n}. Determine the Cartesian product of P and Q.

Solution: The Cartesian product of P and Q is

$$P \times Q = \begin{Bmatrix} (a, k), (a, l), (a, m), (a, n) \\ (b, k), (b, l), (b, m), (b, n) \\ (c, k), (c, l), (c, m), (c, n) \end{Bmatrix}$$

Multisets

A multiset is an unordered collection of elements, in which the multiplicity of an element may be one or more than one or zero. The multiplicity of an element is the number of times the element repeated in the multiset. In other words, we can say that an element can appear any number of times in a set.

Example:

1. A = {l, l, m, m, n, n, n, n}
2. B = {a, a, a, a, a, c}

Operations on Multisets

1. Union of Multisets: The Union of two multisets A and B is a multiset such that the multiplicity of an element is equal to the maximum of the multiplicity of an element in A and B and is denoted by A ∪ B.

Example:

1. Let A = {l, l, m, m, n, n, n, n}
2. B = {l, m, m, m, n},
3. A ∪ B = {l, l, m, m, m, n, n, n, n}

2. Intersections of Multisets: The intersection of two multisets A and B, is a multiset such that the multiplicity of an element is equal to the minimum of the multiplicity of an element in A and B and is denoted by A ∩ B.

Example:

1. Let A = {l, l, m, n, p, q, q, r}

2. B = {l, m, m, p, q, r, r, r, r}
3. A ∩ B = {l, m, p, q, r}.

3. Difference of Multisets: The difference of two multisets A and B, is a multiset such that the multiplicity of an element is equal to the multiplicity of the element in A minus the multiplicity of the element in B if the difference is +ve, and is equal to 0 if the difference is 0 or negative

Example:

1. Let A = {l, m, m, m, n, n, n, p, p, p}
2. B = {l, m, m, m, n, r, r, r}
3. A - B = {n, n, p, p, p}

4. Sum of Multisets: The sum of two multisets A and B, is a multiset such that the multiplicity of an element is equal to the sum of the multiplicity of an element in A and B

Example:

1. Let A = {l, m, n, p, r}
2. B = {l, l, m, n, n, n, p, r, r}
3. A + B = {l, l, l, m, m, n, n, n, n, p, p, r, r, r}

5. Cardinality of Sets: The cardinality of a multiset is the number of distinct elements in a multiset without considering the multiplicity of an element

Example:

1. A = {l, l, m, m, n, n, n, p, p, p, p, q, q, q}

The cardinality of the multiset A is 5.

Ordered Set

It is defined as the ordered collection of distinct objects.

Example:

1. Roll no {3, 6, 7, 8, 9}
2. Week Days {S, M, T, W, W, TH, F, S, S}

Ordered Pairs

An Ordered Pair consists of two elements such that one of them is designated as the first member and other as the second member.

(a, b) and (b, a) are two different ordered pair. An ordered triple can also be written regarding an ordered pair as {(a, b) c}

An ordered Quadrable is an ordered pair {(((a, b), c) d)} with the first element as ordered triple.

An ordered n-tuple is an ordered pair where the first component is an ordered (n - 1) tuple, and the n^{th} element is the second component.

1. {(n -1), n}

Example:

Ordered set of 5 elements

$$\underbrace{\{(((a, b), c),}_{(n-1)}\ d)\ \underset{\overset{\downarrow}{5th}}{e}\}$$

Cardinality and Countability

Definitions:

For now, we will not use the symbol $|X|$ by itself. It does not mean "the number of elements of the set", although our definitions will be consistent with this meaning. Until told otherwise, you may **only** use the symbol $|X|$ as part of the phrases "$|X|\leq|Y||X|\leq|Y|$", "$|X|\geq|Y||X|\geq|Y|$", or "$|X|=|Y||X|=|Y|$". This will prevent you from accidentally using reasoning that only applies to finite sets.

Definitions: Let X and Y be sets.

- $|X| \geq |Y|$ means "there exists a surjection $f:X \rightarrow Y$.
- $|X| \leq |Y|$ means "there exists an injection $f:X \rightarrow Y$.
- $|X| = |Y|$ means "there exists a bijection $f:X \rightarrow Y$

MCS uses the notation "X surj Y" (respectively "inj" or "bij"), but this can be confusing: surjectivity is a property of *functions*, not sets. For example, just because you find one non-injective function from X to Y does **not** mean that $|X| \nless |Y|$.

Properties of cardinality

It is not obvious that this use of ≤ and ≥ is justified. There are many things to prove, most of which are easy. We proved one of them:

Claim: (Cantor-Schroder-Bernstein) $|X| \geq |Y|$ then $|Y| \leq |X|$.

Proof: Suppose $|X| \geq |Y|$. Then by definition, there exists a surjection $f:X \rightarrow Y$. We showed last time that f must have a right inverse $g:Y \rightarrow X$ (which means $f \circ g = id$). This means g has a left inverse (namely f), so g must be injective. Therefore there is an injection from Y to X, so $|Y| \leq |X|$ as required.Other properties you could check for practice:

if $|X| \leq |Y|$ then $|Y| \geq |X|$

if $|X| \leq |Y|$ and $|Y| \leq |Z|$ then $|X| \leq |Z|$

if $|X| = |Y|$ then $|X| \leq |Y|$ and $|X| \geq |Y|$

$|X| = |X|$if $|X| = |Y|$ then $|Y| = |X|$.

There is one property that is true, but the proof is very non-obvious:

Claim: if $|X| \leq |Y|$ and $|Y| \leq |X|$ then $|Y| = |X|$.

The proof is beyond the scope of the course, but it is worth starting it to see why it is hard. If you are curious, here is a proof from a previous semester. You are not responsible for knowing this proof, but it only uses techniques that you should be comfortable with so you should be able to read it.

Note: You may use these properties without proof, unless we ask you to prove them.

Countability

Informally, a set X is countable if you can put it in a (potentially infinite) list. This can be formalized by saying that there is a "first element", a "second element", and so on, and each element is the "nth" element for some n. In other words, there should exist a surjection $f:N\to X$. Even more concisely:

Definition: X is countable if $|N|\geq|X|$.

Equivalently, X is countable if there exists a surjection $f:N\to X$ there exists an injection $f:X\to N$ $|X|\leq|N|$.

If $|X|=|N|$ then we say X is countably infinite.

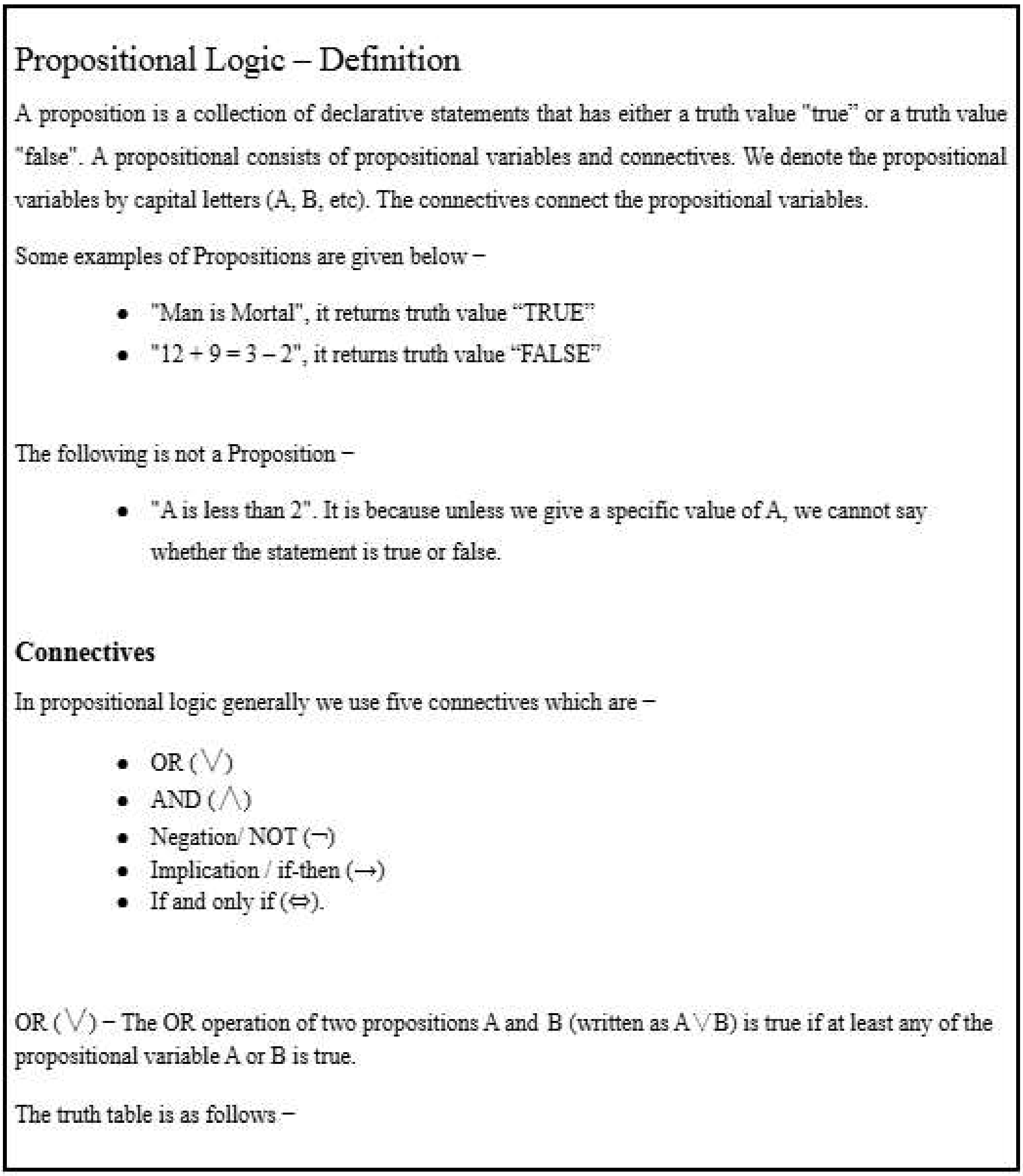
Examples

- the set N is countable; the identity function is a surjection
- the set $X=N\cup\{-1\}$ is countable; let $f:N\to X$ be given by $f(n)=n-1$. You can check that f is surjective
- the set of integers Z is countable. Let $f:N\to Z$ be given by $f(n)::=-n/2$ if n is even and $(n+1)/2$ if n is odd. We must show f is surjective, i.e. for all $i\equiv Z$, there exists $n\equiv N$ such that $f(n)=i$. To do so, choose an arbitrary i. If $i>0$, let $n=2i-1$. We see that n is odd, so $f(n)=(n+1)/2=i$, and thus i is in the image of f. If $i\leq0$ on the other hand, we can choose $n=-2i$. Then n is even, so $f(n)=-(-2i)/2=i$, and again, we see that i is in the image of f. In either case, i is in the image of f, so f must be surjective.
- Next time we will show that Q is countable, but R is uncountable.

Propositional Logic

The rules of mathematical logic specify methods of reasoning mathematical statements. Greek philosopher, Aristotle, was the pioneer of logical reasoning. Logical reasoning provides the theoretical base for many areas of mathematics and consequently computer science. It has many practical applications in computer science like design of computing machines, artificial intelligence, definition of data structures for programming languages etc.

Propositional Logic is concerned with statements to which the truth values, “true” and “false”, can be assigned. The purpose is to analyze these statements either individually or in a composite manner.



Propositional Logic – Definition

A proposition is a collection of declarative statements that has either a truth value "true" or a truth value "false". A propositional consists of propositional variables and connectives. We denote the propositional variables by capital letters (A, B, etc). The connectives connect the propositional variables.

Some examples of Propositions are given below −

- "Man is Mortal", it returns truth value "TRUE"
- "12 + 9 = 3 – 2", it returns truth value "FALSE"

The following is not a Proposition −

- "A is less than 2". It is because unless we give a specific value of A, we cannot say whether the statement is true or false.

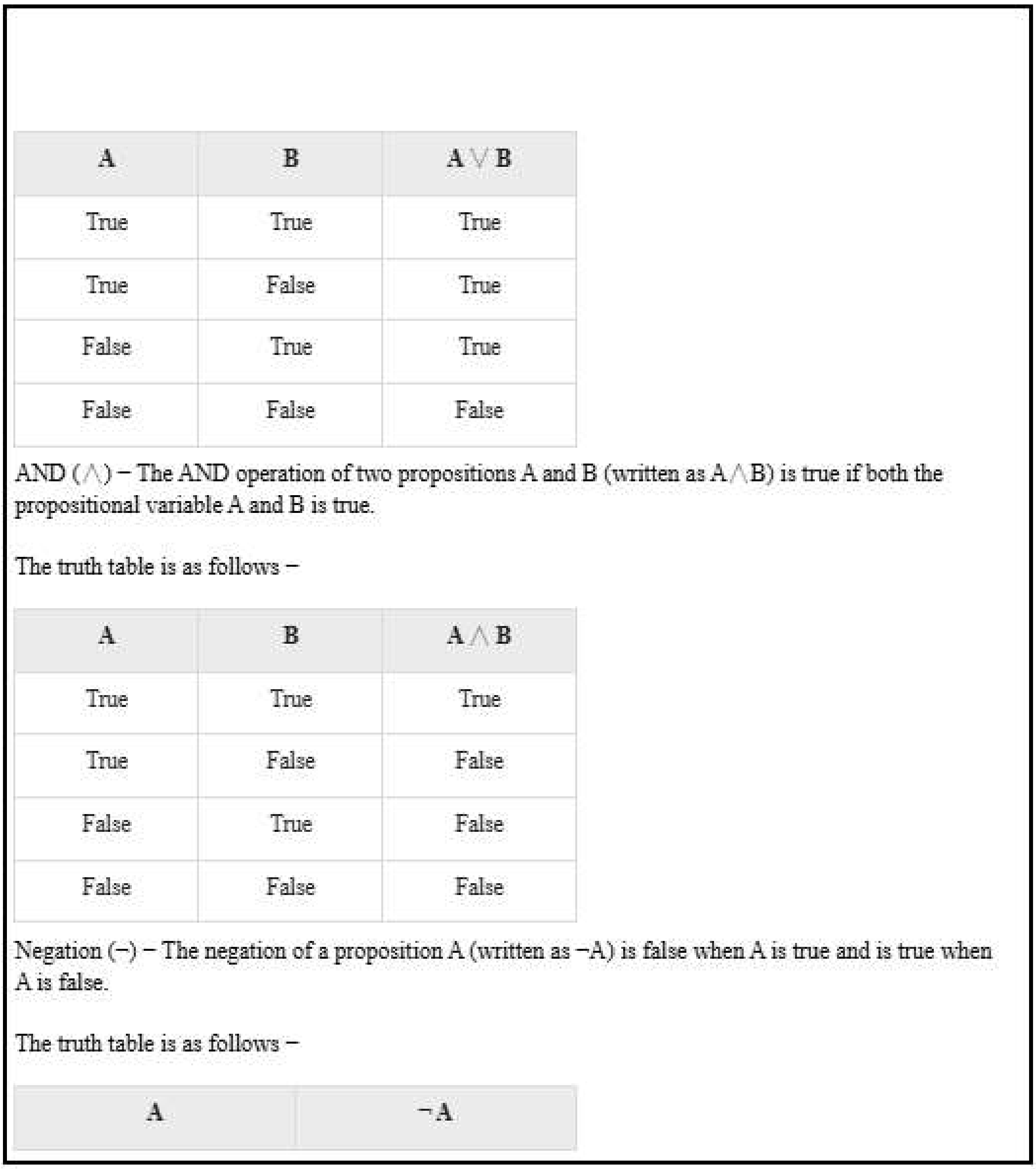
Connectives

In propositional logic generally we use five connectives which are −

- OR (⋁)
- AND (⋀)
- Negation/ NOT (¬)
- Implication / if-then (→)
- If and only if (⇔).

OR (⋁) − The OR operation of two propositions A and B (written as A⋁B) is true if at least any of the propositional variable A or B is true.

The truth table is as follows −



A	B	A ∨ B
True	True	True
True	False	True
False	True	True
False	False	False

AND (∧) − The AND operation of two propositions A and B (written as A∧B) is true if both the propositional variable A and B is true.

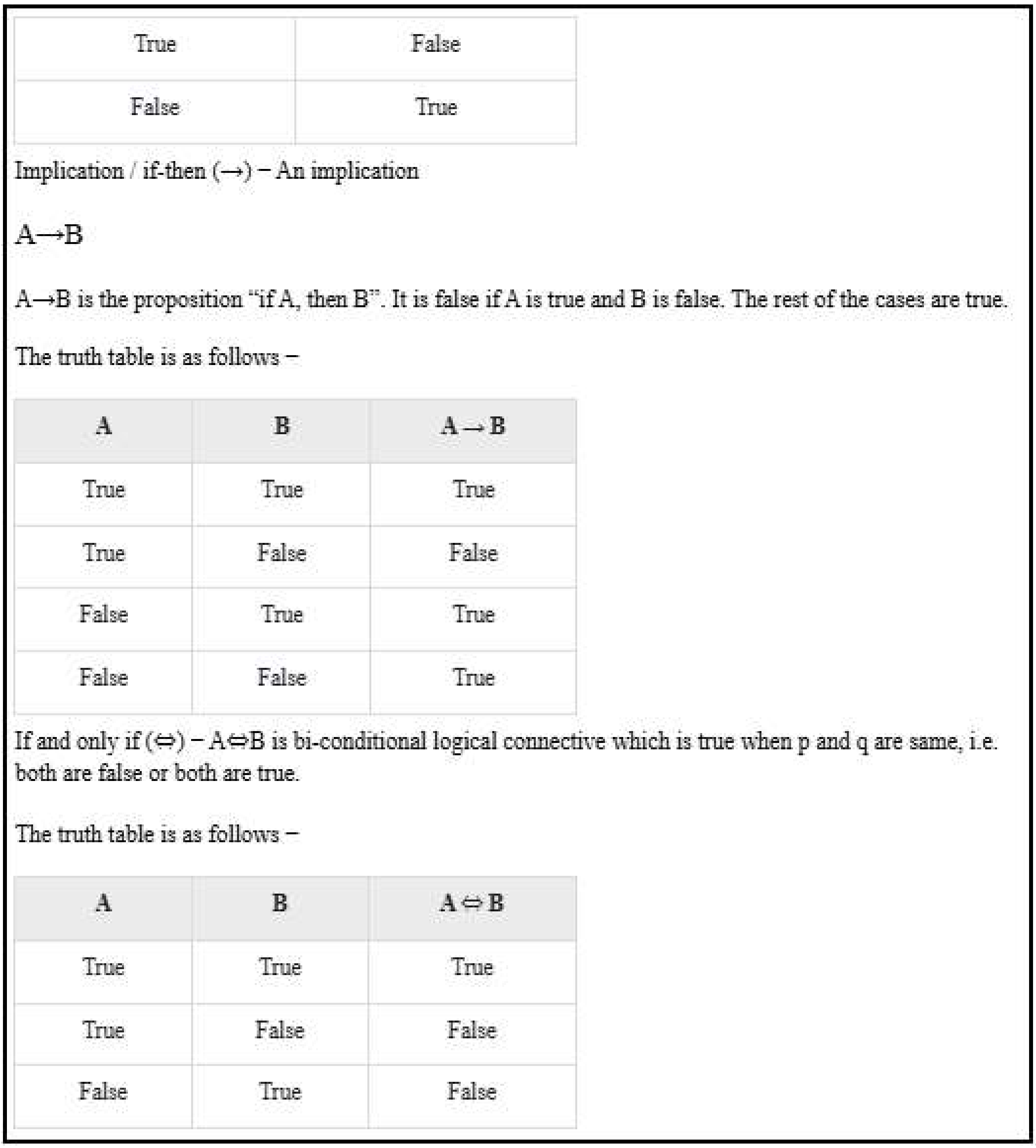
The truth table is as follows −

A	B	A ∧ B
True	True	True
True	False	False
False	True	False
False	False	False

Negation (¬) − The negation of a proposition A (written as ¬A) is false when A is true and is true when A is false.

The truth table is as follows −

A	¬A



True	False
False	True

Implication / if-then (→) − An implication

A→B

A→B is the proposition "if A, then B". It is false if A is true and B is false. The rest of the cases are true.

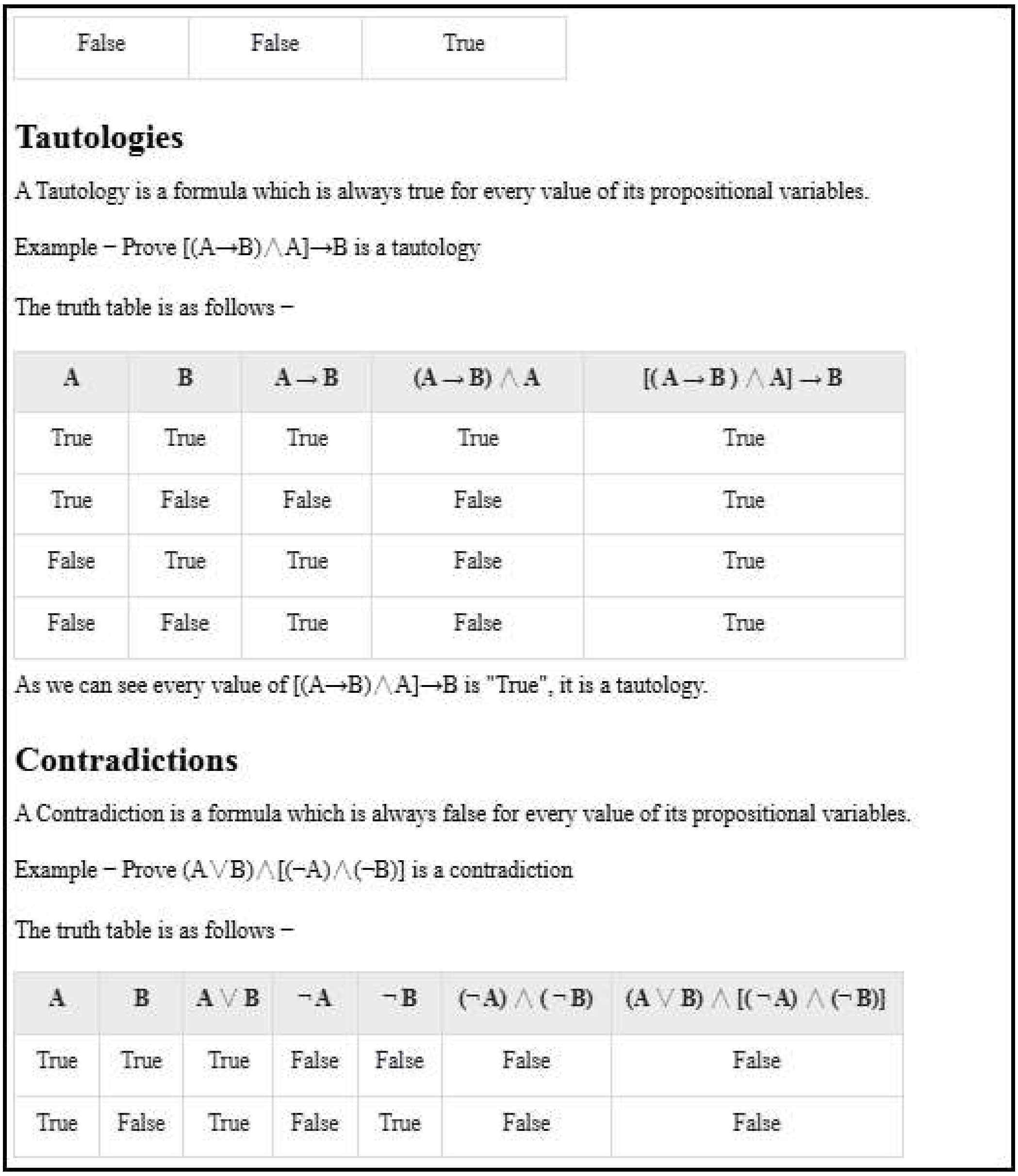
The truth table is as follows −

A	B	A → B
True	True	True
True	False	False
False	True	True
False	False	True

If and only if (⇔) − A⇔B is bi-conditional logical connective which is true when p and q are same, i.e. both are false or both are true.

The truth table is as follows −

A	B	A ⇔ B
True	True	True
True	False	False
False	True	False



False	False	True

Tautologies

A Tautology is a formula which is always true for every value of its propositional variables.

Example − Prove $[(A \rightarrow B) \wedge A] \rightarrow B$ is a tautology

The truth table is as follows −

A	**B**	**$A \rightarrow B$**	**$(A \rightarrow B) \wedge A$**	**$[(A \rightarrow B) \wedge A] \rightarrow B$**
True	True	True	True	True
True	False	False	False	True
False	True	True	False	True
False	False	True	False	True

As we can see every value of $[(A \rightarrow B) \wedge A] \rightarrow B$ is "True", it is a tautology.

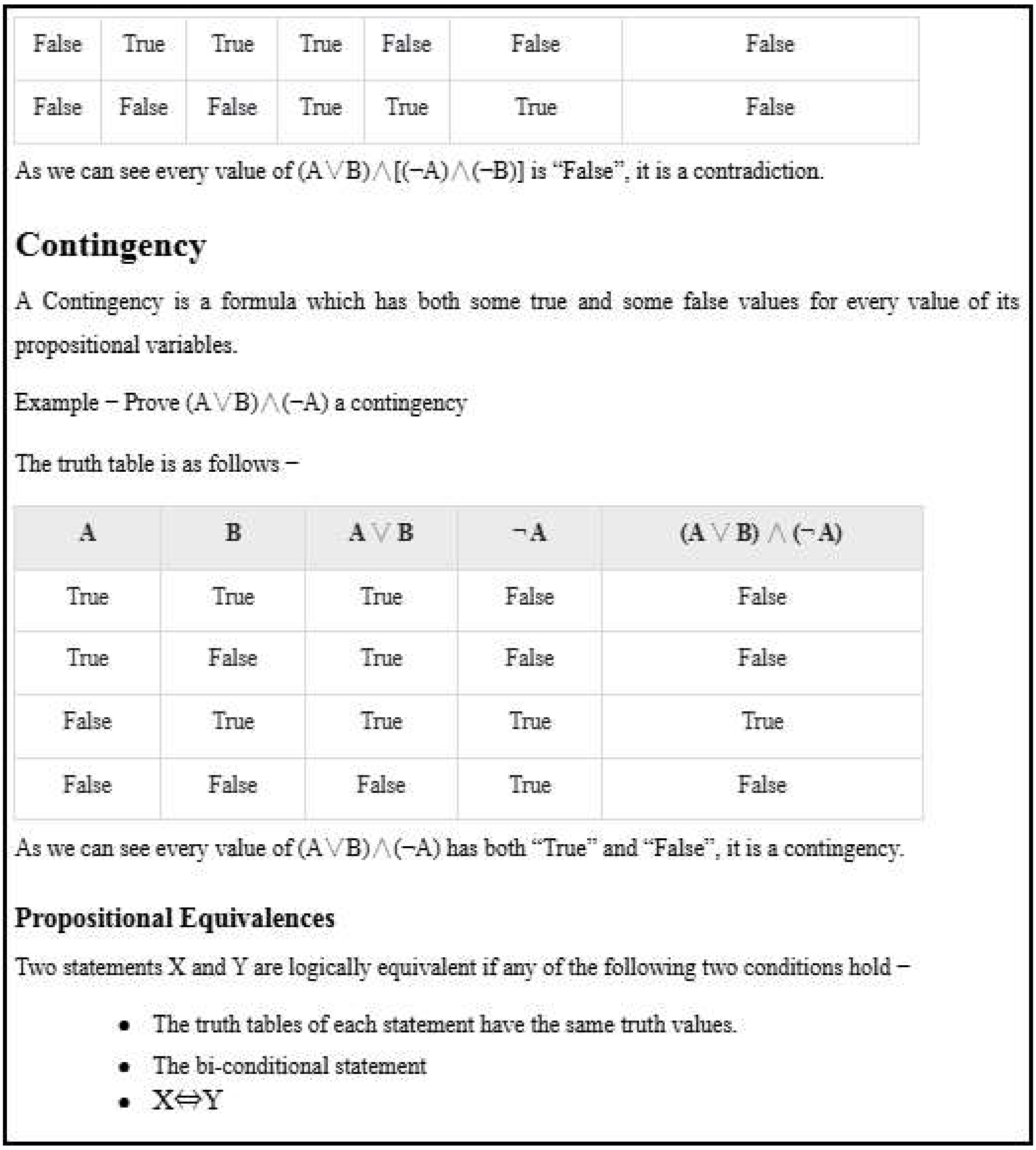
Contradictions

A Contradiction is a formula which is always false for every value of its propositional variables.

Example − Prove $(A \vee B) \wedge [(-A) \wedge (-B)]$ is a contradiction

The truth table is as follows −

A	**B**	**$A \vee B$**	**$\neg A$**	**$\neg B$**	**$(\neg A) \wedge (\neg B)$**	**$(A \vee B) \wedge [(\neg A) \wedge (\neg B)]$**
True	True	True	False	False	False	False
True	False	True	False	True	False	False



False	True	True	True	False	False	False
False	False	False	True	True	True	False

As we can see every value of $(A \vee B) \wedge [(-A) \wedge (-B)]$ is “False”, it is a contradiction.

Contingency

A Contingency is a formula which has both some true and some false values for every value of its propositional variables.

Example − Prove $(A \vee B) \wedge (-A)$ a contingency

The truth table is as follows −

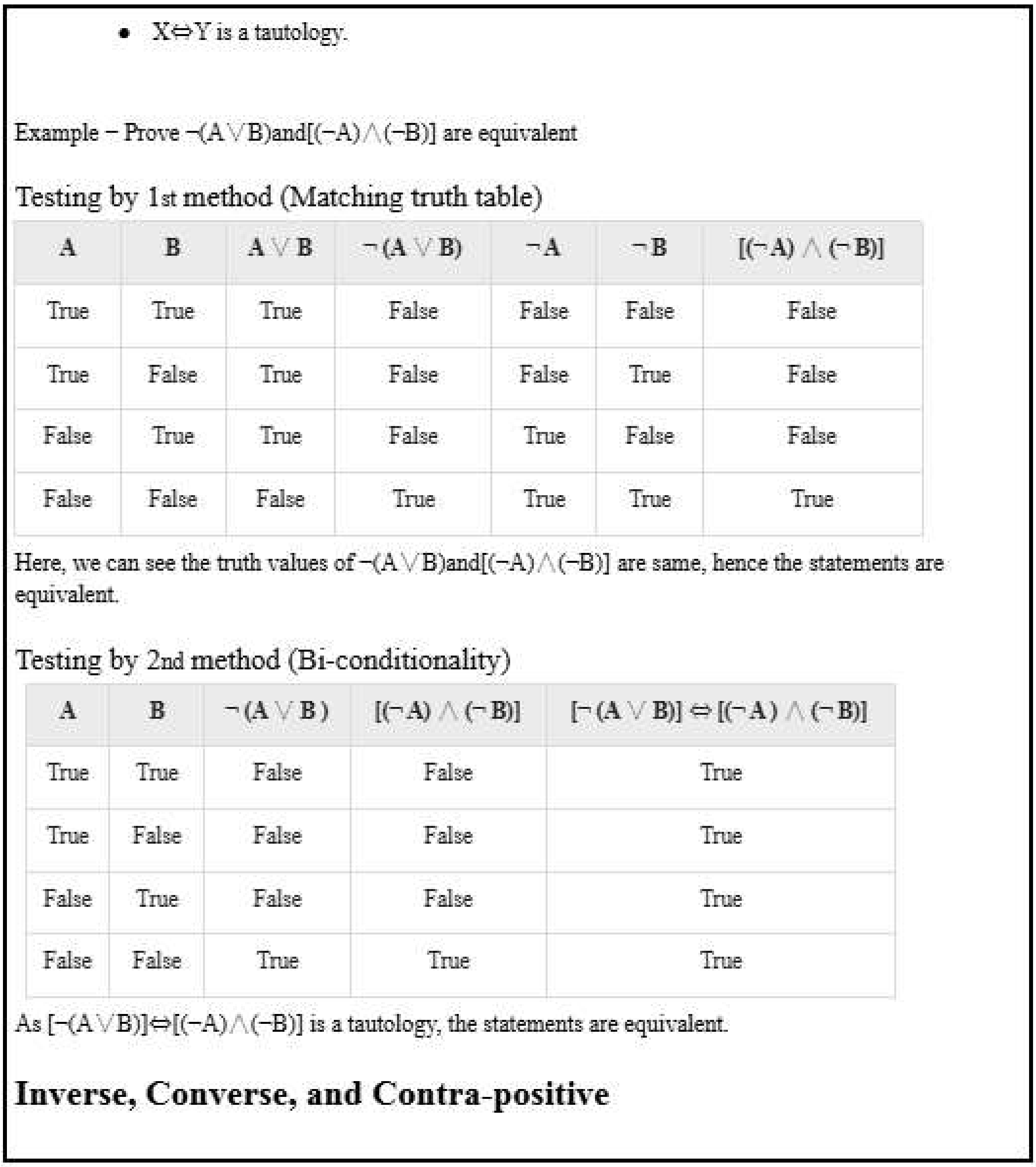
A	B	A ∨ B	¬ A	(A ∨ B) ∧ (¬ A)
True	True	True	False	False
True	False	True	False	False
False	True	True	True	True
False	False	False	True	False

As we can see every value of $(A \vee B) \wedge (-A)$ has both “True” and “False”, it is a contingency.

Propositional Equivalences

Two statements X and Y are logically equivalent if any of the following two conditions hold −

- The truth tables of each statement have the same truth values.
- The bi-conditional statement
- $X \Leftrightarrow Y$



- X⇔Y is a tautology.

Example − Prove ¬(A∨B)and[(−A)∧(−B)] are equivalent

Testing by 1st method (Matching truth table)

A	B	A ∨ B	¬ (A ∨ B)	¬ A	¬ B	[(¬ A) ∧ (¬ B)]
True	True	True	False	False	False	False
True	False	True	False	False	True	False
False	True	True	False	True	False	False
False	False	False	True	True	True	True

Here, we can see the truth values of ¬(A∨B)and[(−A)∧(−B)] are same, hence the statements are equivalent.

Testing by 2nd method (Bi-conditionality)

A	B	¬ (A ∨ B)	[(¬ A) ∧ (¬ B)]	[¬ (A ∨ B)] ⇔ [(¬ A) ∧ (¬ B)]
True	True	False	False	True
True	False	False	False	True
False	True	False	False	True
False	False	True	True	True

As [¬(A∨B)]⇔[(−A)∧(−B)] is a tautology, the statements are equivalent.

Inverse, Converse, and Contra-positive

Implication / if-then (→) is also called a conditional statement. It has two parts –

- Hypothesis, p
- Conclusion, q

As mentioned earlier, it is denoted as $p \to q$.Example of Conditional Statement – "If you do your homework, you will not be punished." Here, "you do your homework" is the hypothesis, p, and "you will not be punished" is the conclusion, q.

Inverse – An inverse of the conditional statement is the negation of both the hypothesis and the conclusion. If the statement is "If p, then q", the inverse will be "If not p, then not q". Thus the inverse of $p \to q$ is $\neg p \to \neg q$.

Example – The inverse of "If you do your homework, you will not be punished" is "If you do not do your homework, you will be punished."

Converse – The converse of the conditional statement is computed by interchanging the hypothesis and the conclusion. If the statement is "If p, then q", the converse will be "If q, then p". The converse of

$p \to q$ is $q \to p$.

Example – The converse of "If you do your homework, you will not be punished" is "If you will not be punished, you do your homework".

Contra-positive – The contra-positive of the conditional is computed by interchanging the hypothesis and the conclusion of the inverse statement. If the statement is "If p, then q", the contra-positive will be "If not q, then not p". The contra-positive of

$p \to q$ is $\neg q \to \neg p$.

Example − The Contra-positive of " If you do your homework, you will not be punished" is "If you are punished, you did not do your homework".

Duality Principle

Duality principle states that for any true statement, the dual statement obtained by interchanging unions into intersections (and vice versa) and interchanging Universal set into Null set (and vice versa) is also true. If dual of any statement is the statement itself, it is said self-dual statement.

Example − The dual of $(A \cap B) \cup C$ is $(A \cup B) \cap C$

Normal Forms

We can convert any proposition in two normal forms −

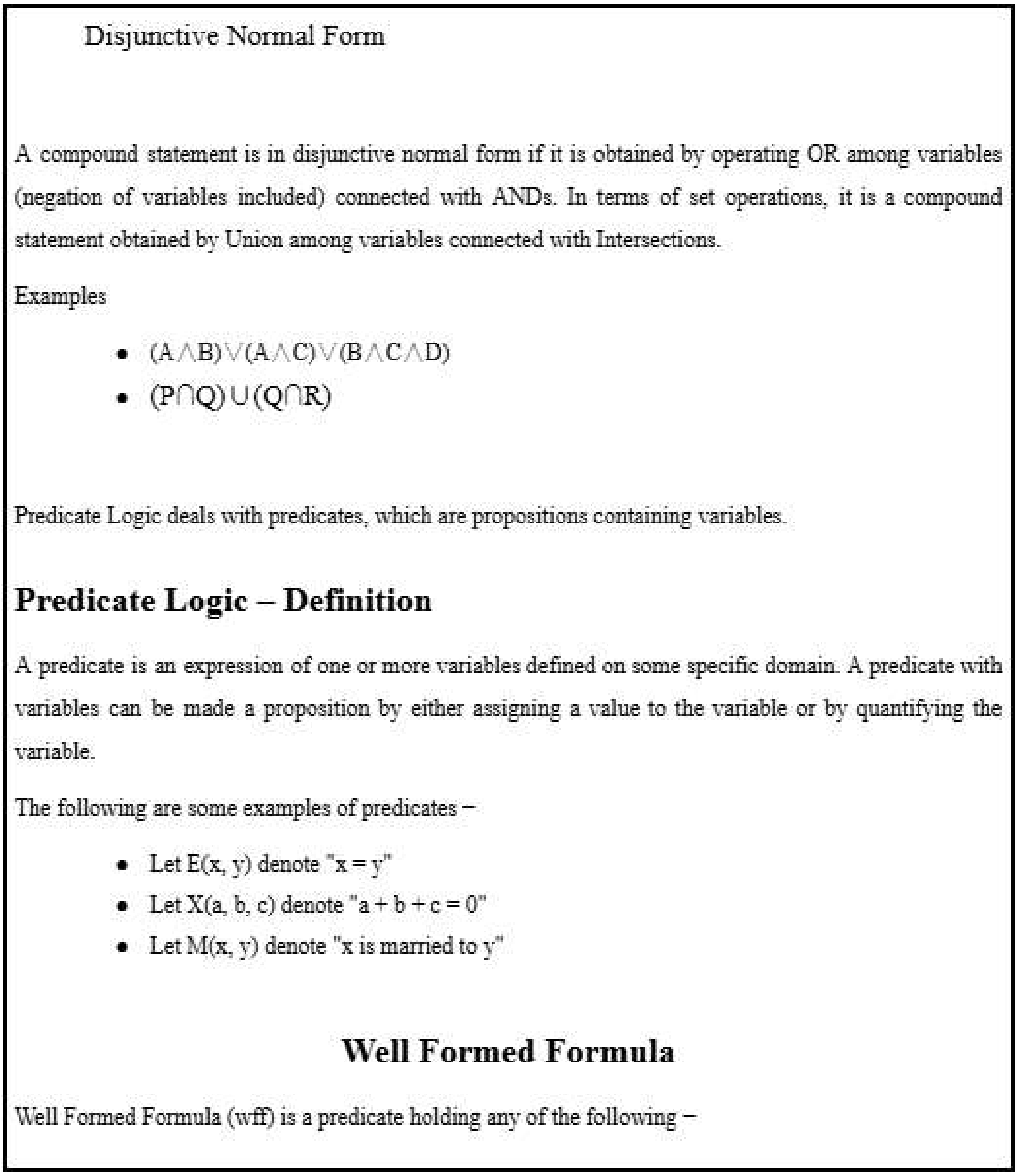
- Conjunctive normal form
- Disjunctive normal form

Conjunctive Normal Form

A compound statement is in conjunctive normal form if it is obtained by operating AND among variables (negation of variables included) connected with ORs. In terms of set operations, it is a compound statement obtained by Intersection among variables connected with Unions.

Examples

- $(A \vee B) \wedge (A \vee C) \wedge (B \vee C \vee D)$
- $(P \cup Q) \cap (Q \cup R)$



Disjunctive Normal Form

A compound statement is in disjunctive normal form if it is obtained by operating OR among variables (negation of variables included) connected with ANDs. In terms of set operations, it is a compound statement obtained by Union among variables connected with Intersections.

Examples

- $(A \wedge B) \vee (A \wedge C) \vee (B \wedge C \wedge D)$
- $(P \cap Q) \cup (Q \cap R)$

Predicate Logic deals with predicates, which are propositions containing variables.

Predicate Logic – Definition

A predicate is an expression of one or more variables defined on some specific domain. A predicate with variables can be made a proposition by either assigning a value to the variable or by quantifying the variable.

The following are some examples of predicates −

- Let E(x, y) denote "x = y"
- Let X(a, b, c) denote "a + b + c = 0"
- Let M(x, y) denote "x is married to y"

Well Formed Formula

Well Formed Formula (wff) is a predicate holding any of the following −

- All propositional constants and propositional variables are wffs
- If x is a variable and Y is a wff,
- $\forall$xY
- $\forall$xY and
- $\exists$xY
- $\exists$xY are also wff
- Truth value and false values are wffs
- Each atomic formula is a wff
- All connectives connecting wffs are wffs

Quantifiers

The variable of predicates is quantified by quantifiers. There are two types of quantifier in predicate logic − Universal Quantifier and Existential Quantifier.

Universal Quantifier

Universal quantifier states that the statements within its scope are true for every value of the specific variable. It is denoted by the symbol $\forall$.

$\forall$xP(x) is read as for every value of x, P(x) is true.

Example − "Man is mortal" can be transformed into the propositional form $\forall$xP(x) where P(x) is the predicate which denotes x is mortal and the universe of discourse is all men.

Existential Quantifier

Existential quantifier states that the statements within its scope are true for some values of the specific variable. It is denoted by the symbol $\exists$. $\exists$xP(x) is read as for some values of x, P(x) is true.

Example − "Some people are dishonest" can be transformed into the propositional form $\exists$xP(x) where P(x) is the predicate which denotes x is dishonest and the universe of discourse is some people.

Nested Quantifiers

If we use a quantifier that appears within the scope of another quantifier, it is called nested quantifier.

Example

- $\forall a \exists b P(x,y)$ where P(a,b) denotes $a+b=0$
- $\forall a \forall b \forall c P(a,b,c)$ where P(a,b) denotes $a+(b+c)=(a+b)+c$

Note – $\forall a \exists b P(x,y) \neq \exists a \forall b P(x,y)$.

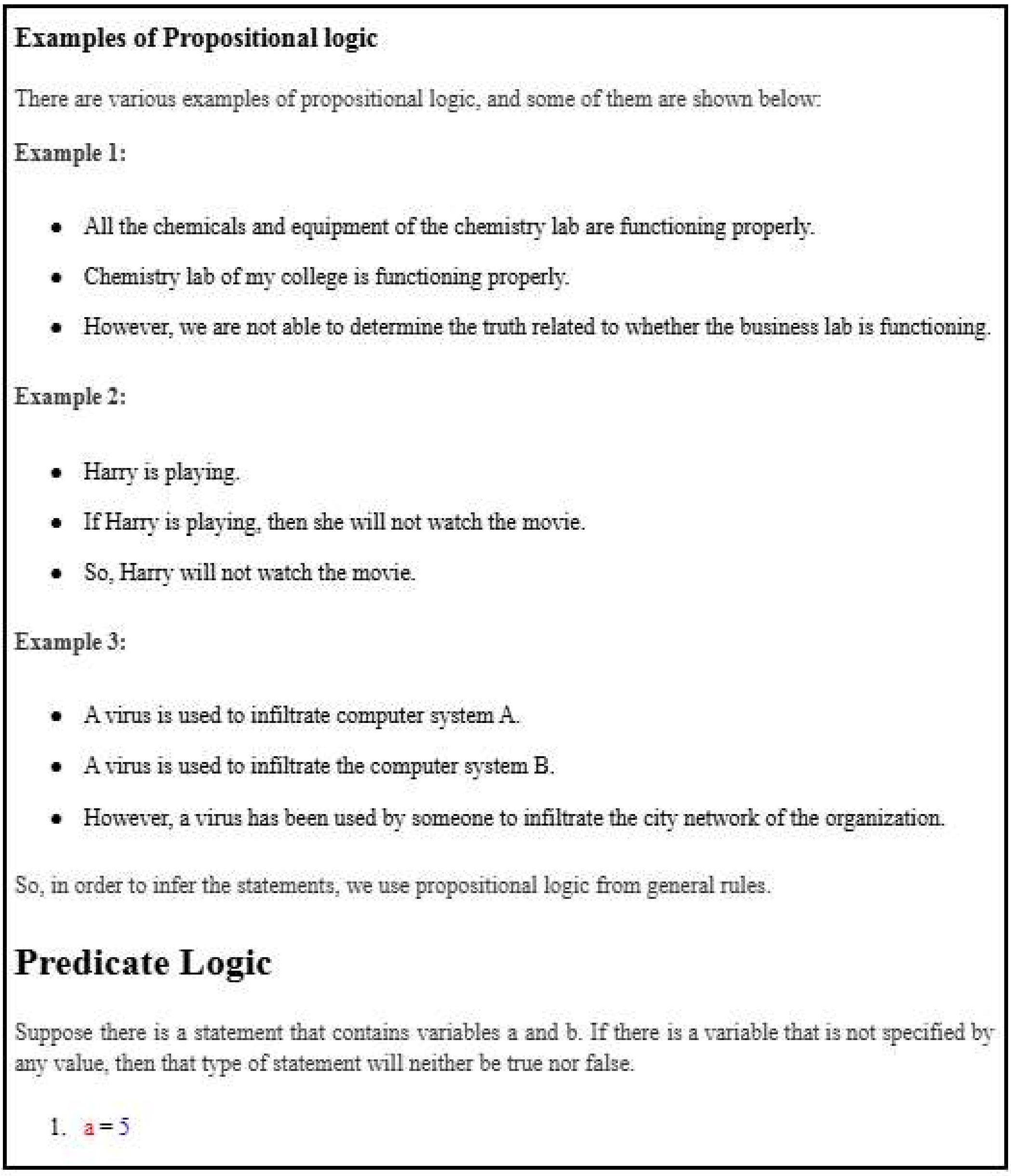
Limitation and Propositional Logic and Predicates

In this section, we will learn about the limitations of Propositional logic and predicates. For this, we will cover the following topics:

- Limitation of Propositional logic
- Predicate Logic and Predicates

Limitations of Propositional Logic

As we know that the propositional logic contains the statements. In case of propositional logic, we are not allowed to conclude the truth of some or ALL statements. Hence, it is not possible to translate or conclude some valid arguments of the propositional logic into purely propositional logic. In case of propositional logic, there is no possibility to describe properties that apply to the object's category. It is also impossible to describe the relationship between those properties.



Examples of Propositional logic

There are various examples of propositional logic, and some of them are shown below:

Example 1:

- All the chemicals and equipment of the chemistry lab are functioning properly.
- Chemistry lab of my college is functioning properly.
- However, we are not able to determine the truth related to whether the business lab is functioning.

Example 2:

- Harry is playing.
- If Harry is playing, then she will not watch the movie.
- So, Harry will not watch the movie.

Example 3:

- A virus is used to infiltrate computer system A.
- A virus is used to infiltrate the computer system B.
- However, a virus has been used by someone to infiltrate the city network of the organization.

So, in order to infer the statements, we use propositional logic from general rules.

Predicate Logic

Suppose there is a statement that contains variables a and b. If there is a variable that is not specified by any value, then that type of statement will neither be true nor false.

```
1. a = 5
```

2. $a = b+4$
3. $a + b = c$

Propositions can be made with the help of predicate logic from statements that have variables. If there is a statement that has a variable, then it will have two parts, which are described as follows:

Suppose there is a statement "a is equal to 5".

- The first part of this statement is "the variable a", which is used to indicate the subject of the statement.
- The second part of this statement is "is equal to 5", which is used to indicate the property that the subject of the statement can have.
- With the help of symbol P(a), we can indicate the statement "a is equal to 5", where P is used to indicate the predicate "is equal to 5", and a is used to indicate the variable.
- Once the variable x is assigned, in this case, statement P(a) becomes the proposition and truth table.

Examples of Predicate

A statement can have one variable or more than one variable. Now we will explain the one variable statement and two variable statements one by one with the help of their examples, which are shown below:

Here we will explain those types of statements that have only one variable. The examples of statements with one variable are described as follows:

Example 1: Suppose there is a statement $P(x) = x>3$. Now we have to determine the truth values of p(4) and p(2).

Solution: From the question, we have a statement $P(x) = x>3$

When we put 2 in place of x, then we will get the following:

- P(2) has a statement "2>;3". This statement is false.
- P(4) has a statement "4>3". This statement is true.

Hence, the truth value of P(2) is false, and the truth value of P(4) is true.

Example 2: Suppose there is a statement P(x) = "A virus is used to infiltrate our computer network". Suppose a virus is used to infiltrate the CS20 and Business. Now we have to determine the truth values of A(CS10), A(CS20), and A(Business).

Solution: From the question, we have a statement:

P(x) = "A virus is used to infiltrate our computer network".

- As we can see that CS10 is not on the infiltrate list. So we can say that A(CS10) will be false.
- The CS20 and Business are on the infiltrate list. So we can say that A(CS20) and A(Business) will be true.

Hence, the truth value of A(CS10) is false, and the truth value of A(CS20) and A(Business) are true.

Two Variables

There can be those types of statements that are related to more than one variable. The examples of statements with two variables are described as follows:

Example 1: Suppose we have a proposition Q(a, b) that has a statement "a = b+6". Now we have to determine the truth value of Q(3, 6) and Q(6, 0).

Solution: From the question, we have a statement

Q(a, b) = "a = b+6".

- Q(3, 6) has a statement "3 = 6 + 6". This statement is false because 3 is not equal to 12.
- Q(6, 0) has a statement "6 = 0 + 6". This statement is true because 6 = 6.

Hence, the truth value of Q(3, 6) is false, and the truth value of Q(6, 0) is true.

Example 2: Suppose we have a proposition Q(a, b) that has a statement "a = b-5". Now we have to determine the truth value of Q(7, 4) and Q(0, 5).

Solution: From the question, we have a statement

Q(a, b) = "a = b-5".

- Q(7, 4) has a statement "7 = 4 - 5". This statement is false because 7 is not equal to -1.
- Q(0, 5) has a statement "0 = 5 - 5". This statement is true because 0 = 0.

Hence, the truth value of Q(7, 4) is false, and the truth value of Q(0, 5) is true.

Quantifiers

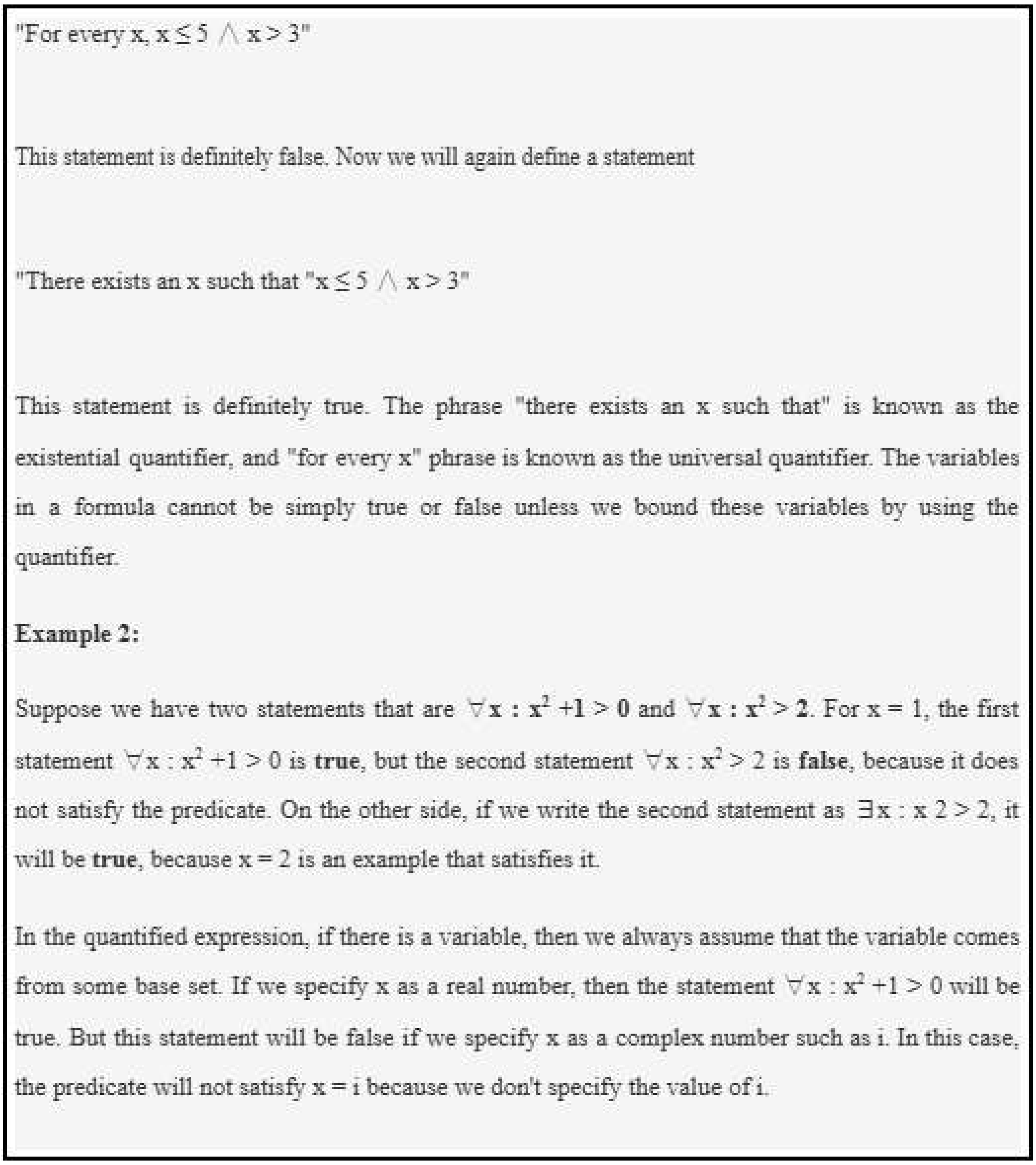
Quantifier is used to quantify the variable of predicates. It contains a formula, which is a type of statement whose truth value may depend on values of some variables. When we assign a fixed value to a predicate, then it becomes a proposition. In another way, we can say that if we quantify the predicate, then the predicate will become a proposition. So quantify is a type of word which refers to quantifies like **"all"** or **"some"**.

There are mainly two types of quantifiers that are universal quantifiers and existential quantifiers. Besides this, we also have other types of quantifiers such as nested quantifiers and Quantifiers in Standard English Usages. Quantifier is mainly used to show that for how many elements, a described predicate is true. It also shows that for all possible values or for some value(s) in the universe of discourse, the predicate is true or not.

Example 1:

"$x \leq 5 \wedge x > 3$"

This statement is false for x= 6 and true for x = 4. Now we will compare the above statement with the following statement



"For every x, $x \leq 5 \land x > 3$"

This statement is definitely false. Now we will again define a statement

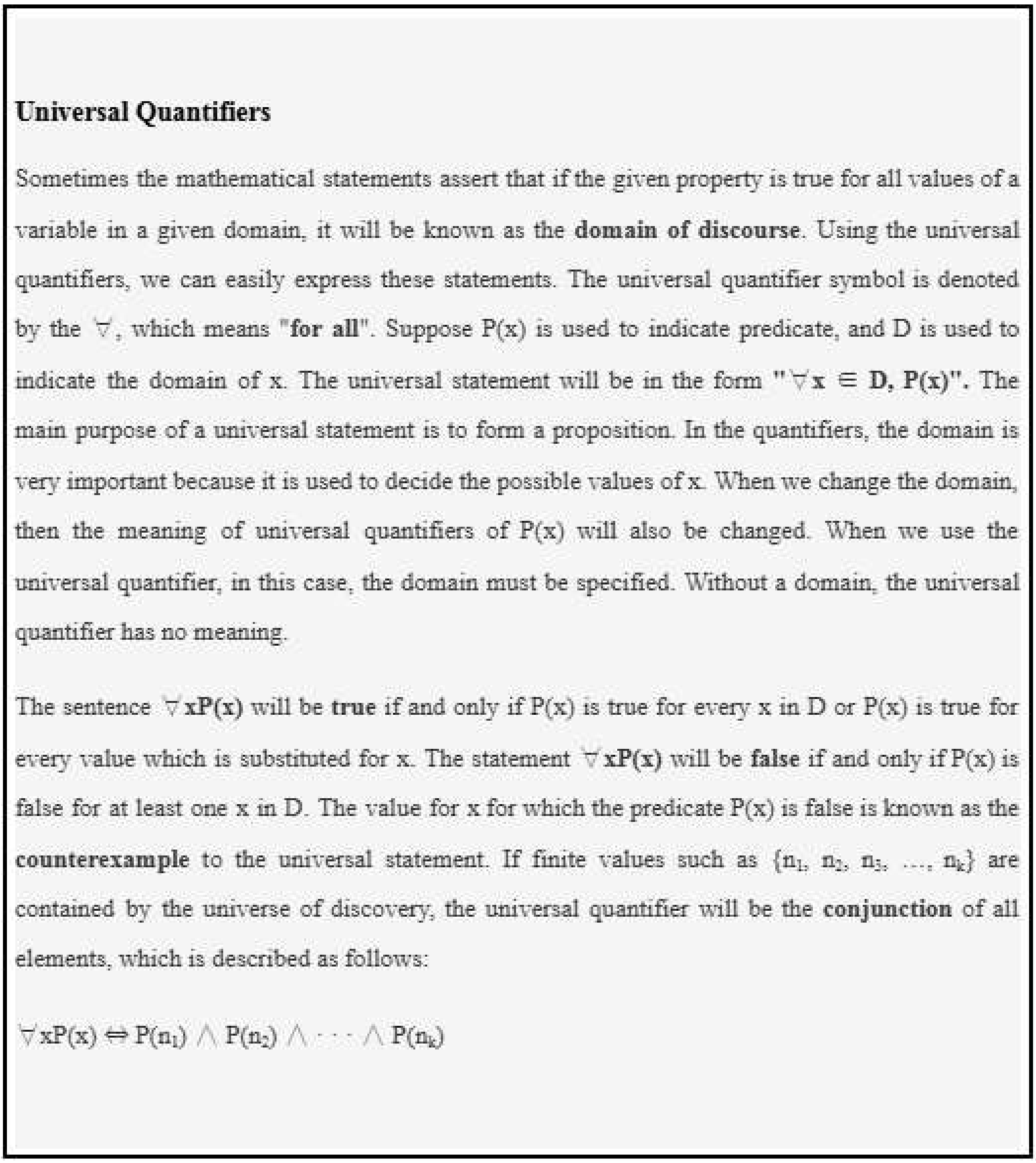
"There exists an x such that "$x \leq 5 \land x > 3$"

This statement is definitely true. The phrase "there exists an x such that" is known as the existential quantifier, and "for every x" phrase is known as the universal quantifier. The variables in a formula cannot be simply true or false unless we bound these variables by using the quantifier.

Example 2:

Suppose we have two statements that are $\forall \mathbf{x} : \mathbf{x}^2 + \mathbf{1} > \mathbf{0}$ and $\forall \mathbf{x} : \mathbf{x}^2 > \mathbf{2}$. For x = 1, the first statement $\forall x : x^2 +1 > 0$ is **true**, but the second statement $\forall x : x^2 > 2$ is **false**, because it does not satisfy the predicate. On the other side, if we write the second statement as $\exists x : x\ 2 > 2$, it will be **true**, because x = 2 is an example that satisfies it.

In the quantified expression, if there is a variable, then we always assume that the variable comes from some base set. If we specify x as a real number, then the statement $\forall x : x^2 +1 > 0$ will be true. But this statement will be false if we specify x as a complex number such as i. In this case, the predicate will not satisfy x = i because we don't specify the value of i.

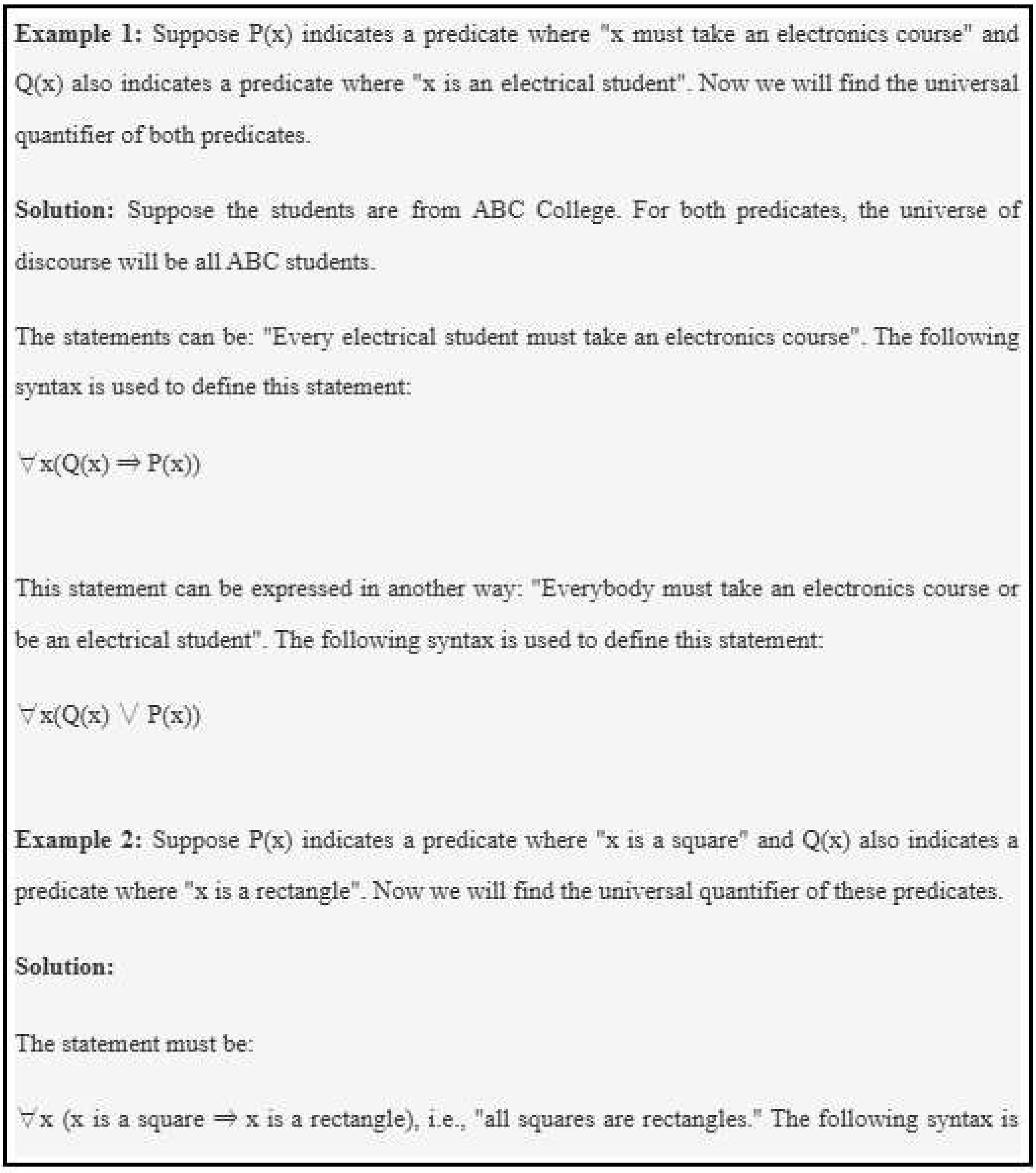


Universal Quantifiers

Sometimes the mathematical statements assert that if the given property is true for all values of a variable in a given domain, it will be known as the **domain of discourse**. Using the universal quantifiers, we can easily express these statements. The universal quantifier symbol is denoted by the $\forall$, which means "**for all**". Suppose P(x) is used to indicate predicate, and D is used to indicate the domain of x. The universal statement will be in the form "**$\forall x \in D, P(x)$**". The main purpose of a universal statement is to form a proposition. In the quantifiers, the domain is very important because it is used to decide the possible values of x. When we change the domain, then the meaning of universal quantifiers of P(x) will also be changed. When we use the universal quantifier, in this case, the domain must be specified. Without a domain, the universal quantifier has no meaning.

The sentence **$\forall xP(x)$** will be **true** if and only if P(x) is true for every x in D or P(x) is true for every value which is substituted for x. The statement **$\forall xP(x)$** will be **false** if and only if P(x) is false for at least one x in D. The value for x for which the predicate P(x) is false is known as the **counterexample** to the universal statement. If finite values such as $\{n_1, n_2, n_3, \ldots, n_k\}$ are contained by the universe of discovery, the universal quantifier will be the **conjunction** of all elements, which is described as follows:

$\forall xP(x) \Leftrightarrow P(n_1) \wedge P(n_2) \wedge \cdots \wedge P(n_k)$



Example 1: Suppose P(x) indicates a predicate where "x must take an electronics course" and Q(x) also indicates a predicate where "x is an electrical student". Now we will find the universal quantifier of both predicates.

Solution: Suppose the students are from ABC College. For both predicates, the universe of discourse will be all ABC students.

The statements can be: "Every electrical student must take an electronics course". The following syntax is used to define this statement:

$\forall x(Q(x) \Rightarrow P(x))$

This statement can be expressed in another way: "Everybody must take an electronics course or be an electrical student". The following syntax is used to define this statement:

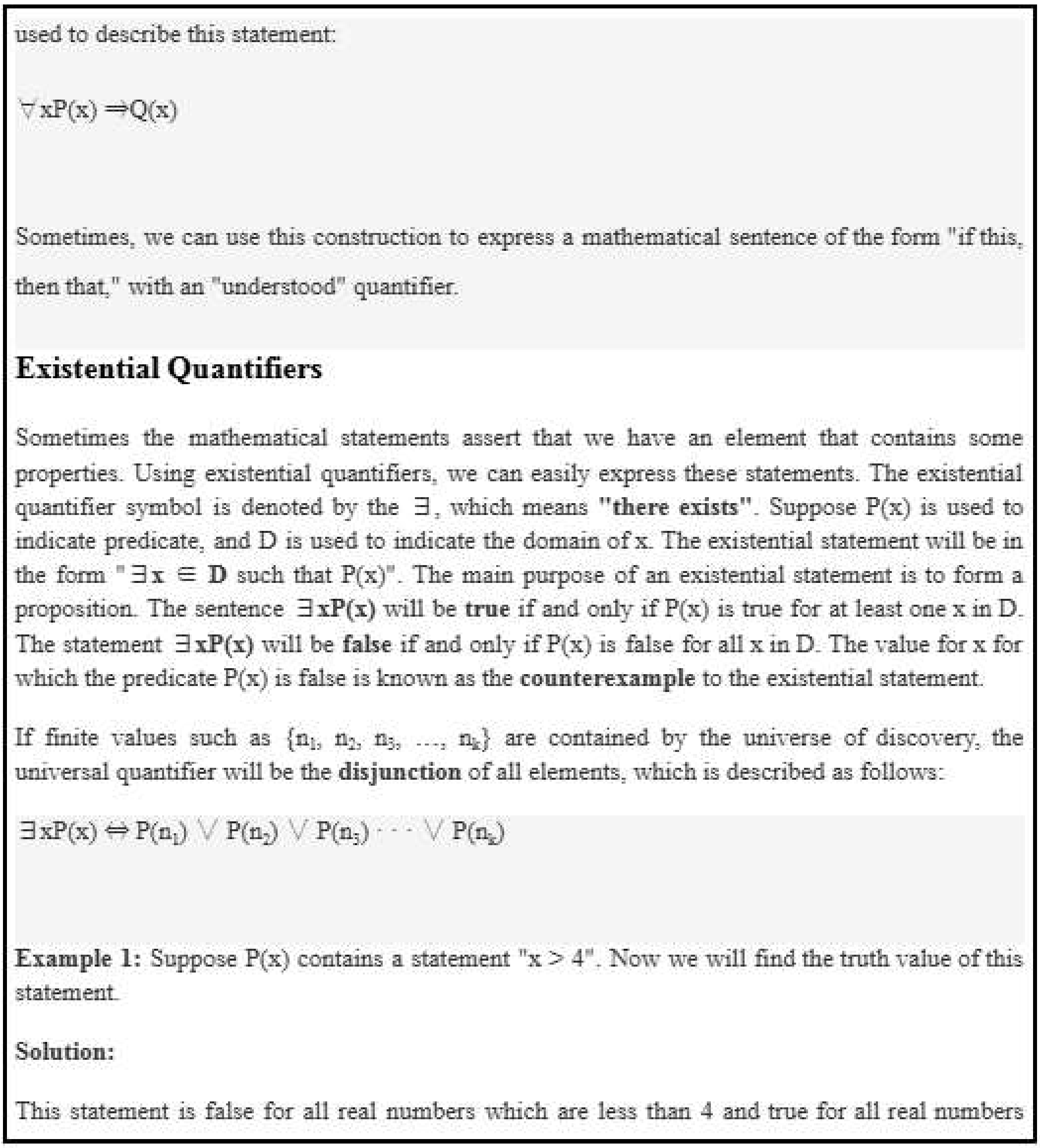
$\forall x(Q(x) \vee P(x))$

Example 2: Suppose P(x) indicates a predicate where "x is a square" and Q(x) also indicates a predicate where "x is a rectangle". Now we will find the universal quantifier of these predicates.

Solution:

The statement must be:

$\forall x$ (x is a square $\Rightarrow$ x is a rectangle), i.e., "all squares are rectangles." The following syntax is



used to describe this statement:

$\forall x P(x) \Rightarrow Q(x)$

Sometimes, we can use this construction to express a mathematical sentence of the form "if this, then that," with an "understood" quantifier.

Existential Quantifiers

Sometimes the mathematical statements assert that we have an element that contains some properties. Using existential quantifiers, we can easily express these statements. The existential quantifier symbol is denoted by the $\exists$, which means **"there exists"**. Suppose P(x) is used to indicate predicate, and D is used to indicate the domain of x. The existential statement will be in the form "**$\exists x \in D$** such that P(x)". The main purpose of an existential statement is to form a proposition. The sentence **$\exists x P(x)$** will be **true** if and only if P(x) is true for at least one x in D. The statement **$\exists x P(x)$** will be **false** if and only if P(x) is false for all x in D. The value for x for which the predicate P(x) is false is known as the **counterexample** to the existential statement.

If finite values such as $\{n_1, n_2, n_3, \ldots, n_k\}$ are contained by the universe of discovery, the universal quantifier will be the **disjunction** of all elements, which is described as follows:

$\exists x P(x) \Leftrightarrow P(n_1) \vee P(n_2) \vee P(n_3) \cdot \cdot \cdot \vee P(n_k)$

Example 1: Suppose P(x) contains a statement "x > 4". Now we will find the truth value of this statement.

Solution:

This statement is false for all real numbers which are less than 4 and true for all real numbers which are greater than 4.

This statement is false for x= 6 and true for x = 4. Now we will compare the above statement with the following statement. So

$\exists x P(x)$ is true

Prime Number in Discrete Mathematics

Overview

An integer $p > 1$ will be known as the prime or prime number if and only if 1 and p are the only positive divisor of p. In simple terms, any number will be known as a prime number if it is divided only by 1 and itself. There is also a term known as composite that will occur if an integer $q > 1$ is not prime. That means if a number is divided by any other number, then it will be known as composite. In other words we can say that if a number is not prime then it will be composite.

For example:

The integers 4, 6, 8, 9 are called composite, and the integers 2, 3, 5, 7, and 11 are called the prime numbers.

Theorem 1:

An integer $p > 1$ will be known as a prime if for all integers x and y, p divides xy. This statement means that for a prime number, p either divides x or y.

Theorem 2:

Every integer n > = 2 must contain a prime factor.

Theorem 3:

Suppose a composite integer is shown by n. In this case, n must have a prime factor that will not exceed $\sqrt{n}$.

Examples of prime numbers are described as follows:

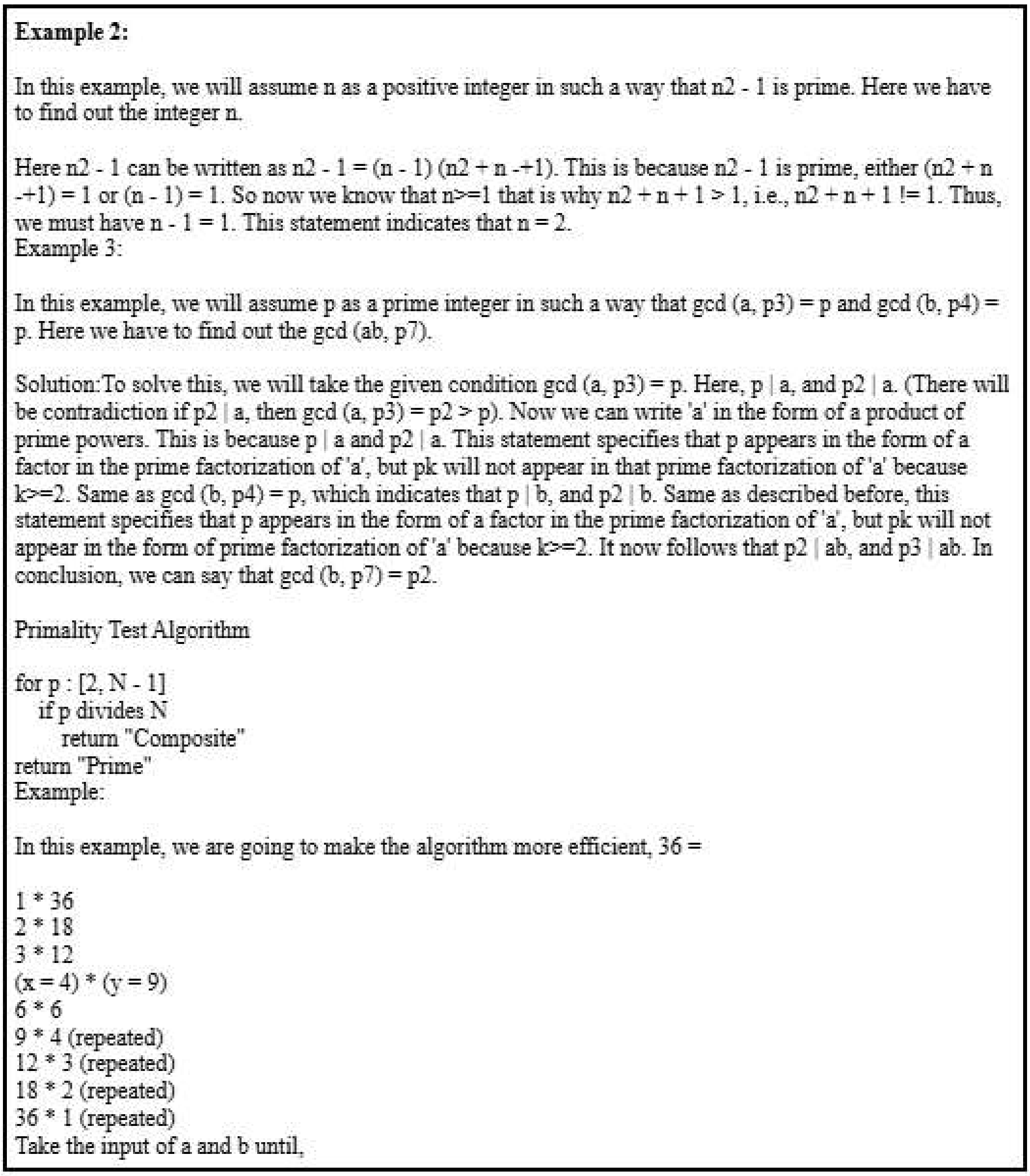
Example 1:

In this example, we have two integers, and we have to determine which one is prime. The first integer is 293, and the second integer is 9823.

Solution:

First, we will find all the primes p in such a way that p2 <= 293. All of these primes are 2, 3, 5, 7, 11, 13, and 17. Now, 293 is not divided by any of these primes. So we can say that 293 is prime.

Now we will find primes p in such a way that p2 <= 9823. All of these primes are 2, 3, 5, 7, 11, 13, 17, etc. Now, 9823 is not divided by any of 2, 3, 5, 7, but 11 divides 9823. So we can say that 9823 is not a prime.



Example 2:

In this example, we will assume n as a positive integer in such a way that n2 - 1 is prime. Here we have to find out the integer n.

Here n2 - 1 can be written as n2 - 1 = (n - 1) (n2 + n -+1). This is because n2 - 1 is prime, either (n2 + n -+1) = 1 or (n - 1) = 1. So now we know that n>=1 that is why n2 + n + 1 > 1, i.e., n2 + n + 1 != 1. Thus, we must have n - 1 = 1. This statement indicates that n = 2.
Example 3:

In this example, we will assume p as a prime integer in such a way that gcd (a, p3) = p and gcd (b, p4) = p. Here we have to find out the gcd (ab, p7).

Solution:To solve this, we will take the given condition gcd (a, p3) = p. Here, p | a, and p2 | a. (There will be contradiction if p2 | a, then gcd (a, p3) = p2 > p). Now we can write 'a' in the form of a product of prime powers. This is because p | a and p2 | a. This statement specifies that p appears in the form of a factor in the prime factorization of 'a', but pk will not appear in that prime factorization of 'a' because k>=2. Same as gcd (b, p4) = p, which indicates that p | b, and p2 | b. Same as described before, this statement specifies that p appears in the form of a factor in the prime factorization of 'a', but pk will not appear in the form of prime factorization of 'a' because k>=2. It now follows that p2 | ab, and p3 | ab. In conclusion, we can say that gcd (b, p7) = p2.

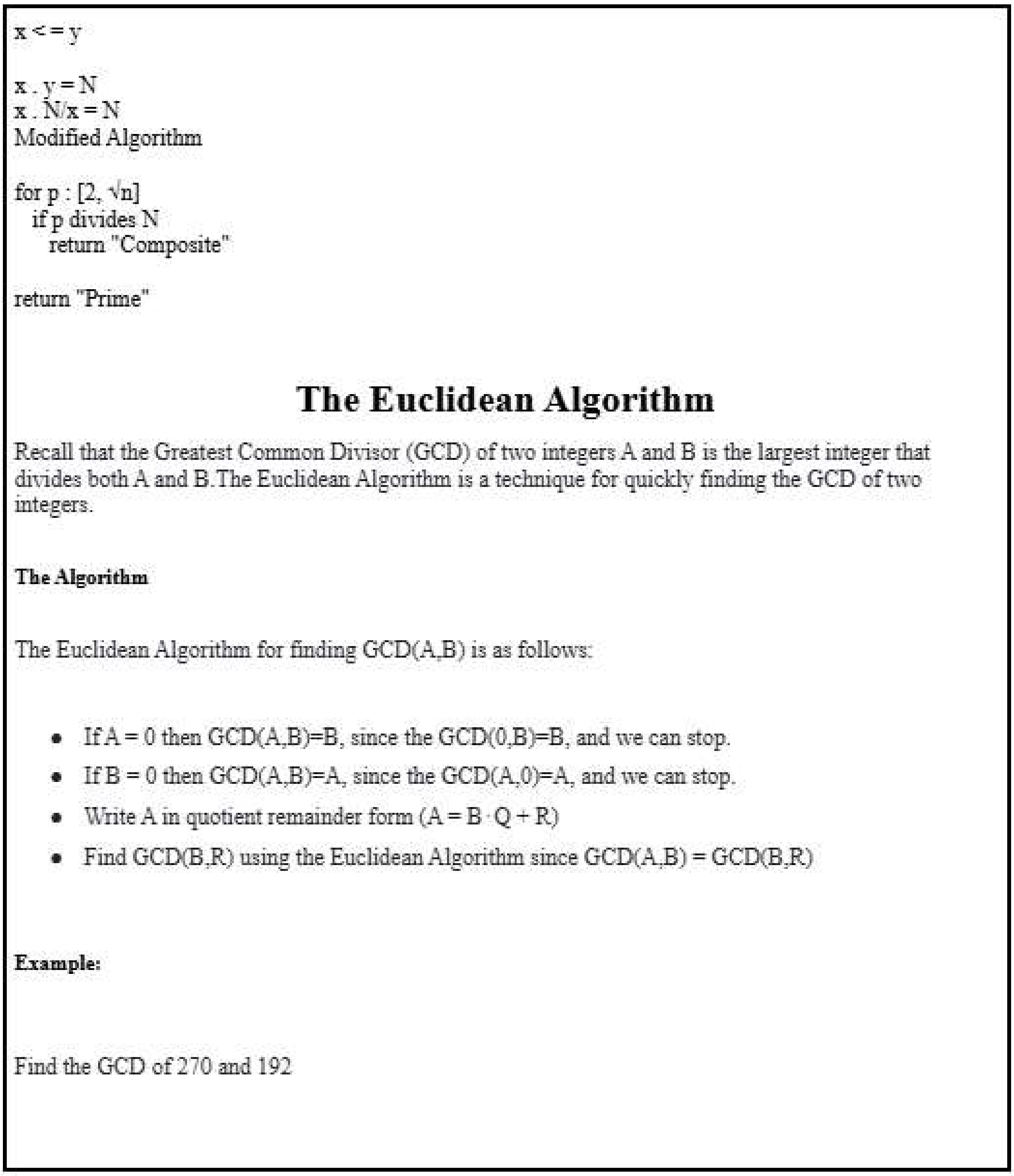
Primality Test Algorithm

```
for p : [2, N - 1]
  if p divides N
     return "Composite"
return "Prime"
```

Example:

In this example, we are going to make the algorithm more efficient, 36 =

1 * 36
2 * 18
3 * 12
(x = 4) * (y = 9)
6 * 6
9 * 4 (repeated)
12 * 3 (repeated)
18 * 2 (repeated)
36 * 1 (repeated)
Take the input of a and b until,



x < = y

x . y = N
x . N/x = N
Modified Algorithm

```
for p : [2, √n]
  if p divides N
    return "Composite"

return "Prime"
```

The Euclidean Algorithm

Recall that the Greatest Common Divisor (GCD) of two integers A and B is the largest integer that divides both A and B.The Euclidean Algorithm is a technique for quickly finding the GCD of two integers.

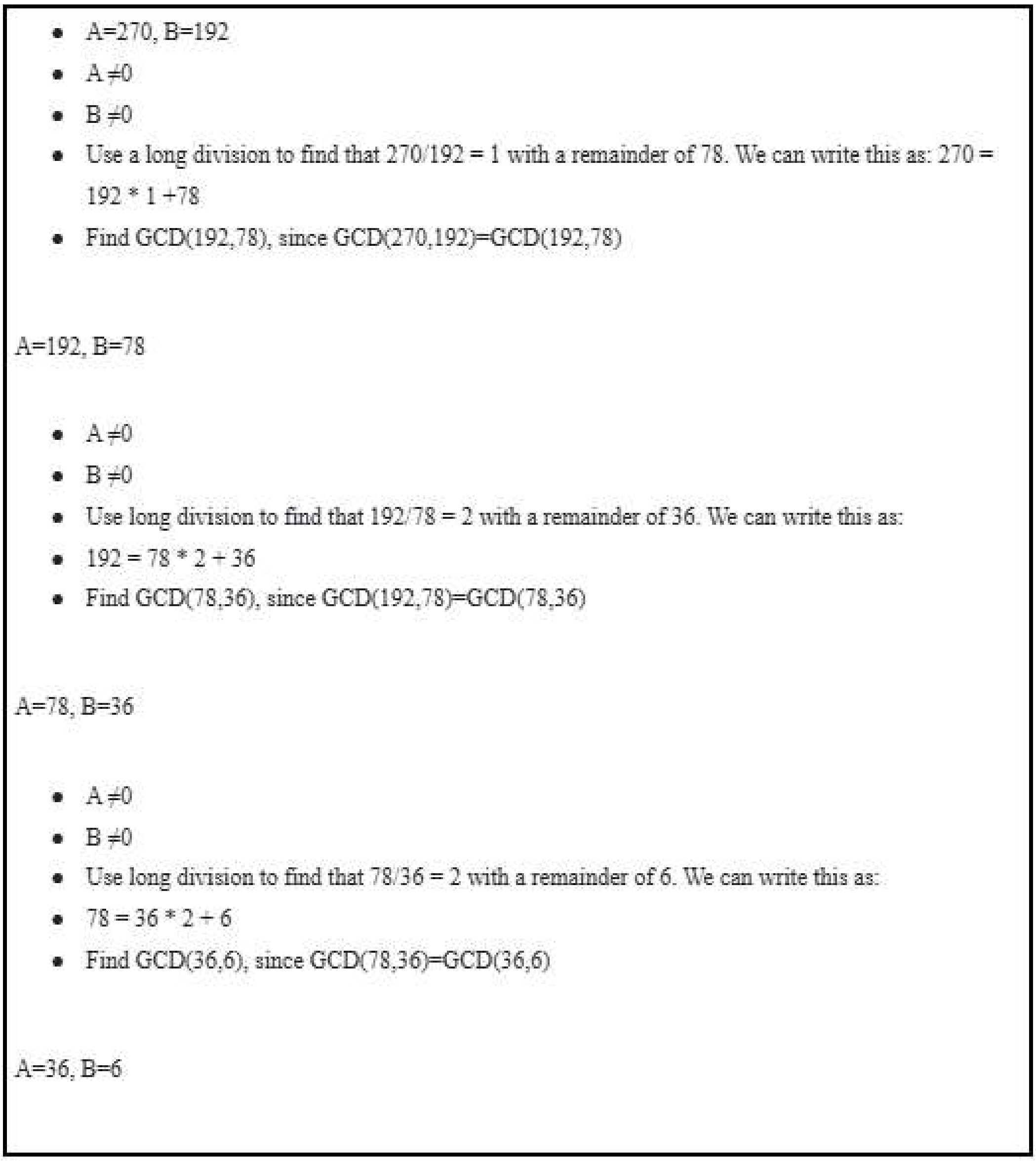
The Algorithm

The Euclidean Algorithm for finding GCD(A,B) is as follows:

- If A = 0 then GCD(A,B)=B, since the GCD(0,B)=B, and we can stop.
- If B = 0 then GCD(A,B)=A, since the GCD(A,0)=A, and we can stop.
- Write A in quotient remainder form (A = B ⋅ Q + R)
- Find GCD(B,R) using the Euclidean Algorithm since GCD(A,B) = GCD(B,R)

Example:

Find the GCD of 270 and 192



- A=270, B=192
- A ≠0
- B ≠0
- Use a long division to find that 270/192 = 1 with a remainder of 78. We can write this as: 270 = 192 * 1 +78
- Find GCD(192,78), since GCD(270,192)=GCD(192,78)

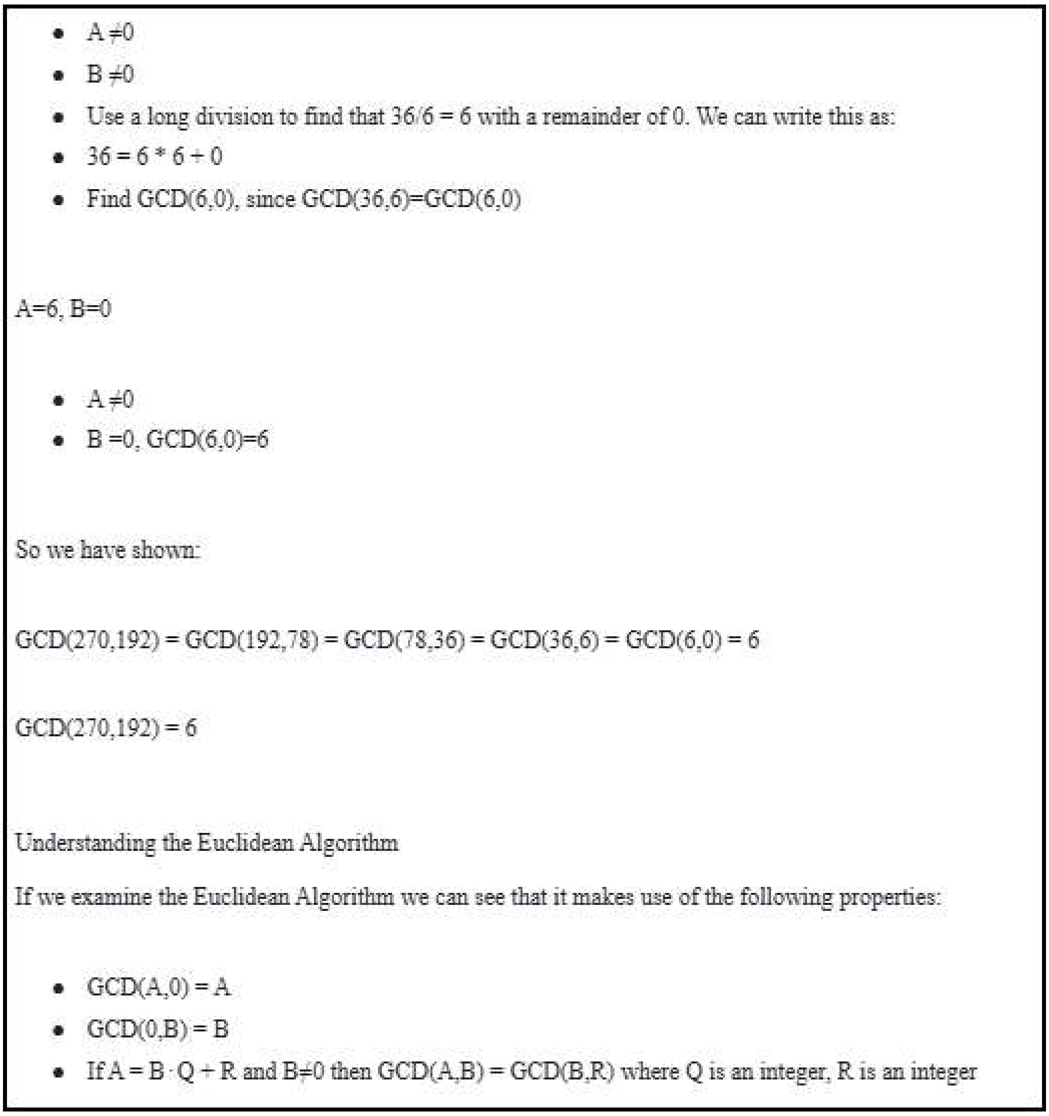
A=192, B=78

- A ≠0
- B ≠0
- Use long division to find that 192/78 = 2 with a remainder of 36. We can write this as:
- 192 = 78 * 2 + 36
- Find GCD(78,36), since GCD(192,78)=GCD(78,36)

A=78, B=36

- A ≠0
- B ≠0
- Use long division to find that 78/36 = 2 with a remainder of 6. We can write this as:
- 78 = 36 * 2 + 6
- Find GCD(36,6), since GCD(78,36)=GCD(36,6)

A=36, B=6



- A ≠0
- B ≠0
- Use a long division to find that 36/6 = 6 with a remainder of 0. We can write this as:
- 36 = 6 * 6 + 0
- Find GCD(6,0), since GCD(36,6)=GCD(6,0)

A=6, B=0

- A ≠0
- B =0, GCD(6,0)=6

So we have shown:

GCD(270,192) = GCD(192,78) = GCD(78,36) = GCD(36,6) = GCD(6,0) = 6

GCD(270,192) = 6

Understanding the Euclidean Algorithm

If we examine the Euclidean Algorithm we can see that it makes use of the following properties:

- GCD(A,0) = A
- GCD(0,B) = B
- If $A = B \cdot Q + R$ and $B \neq 0$ then GCD(A,B) = GCD(B,R) where Q is an integer, R is an integer

between 0 and B-1

The first two properties let us find the GCD if either number is 0. The third property lets us take a larger, more difficult to solve problem, and reduce it to a smaller, easier to solve problem.

The Euclidean Algorithm makes use of these properties by rapidly reducing the problem into easier and easier problems, using the third property, until it is easily solved by using one of the first two properties.

We can understand why these properties work by proving them.

We can prove that GCD(A,0)=A is as follows:

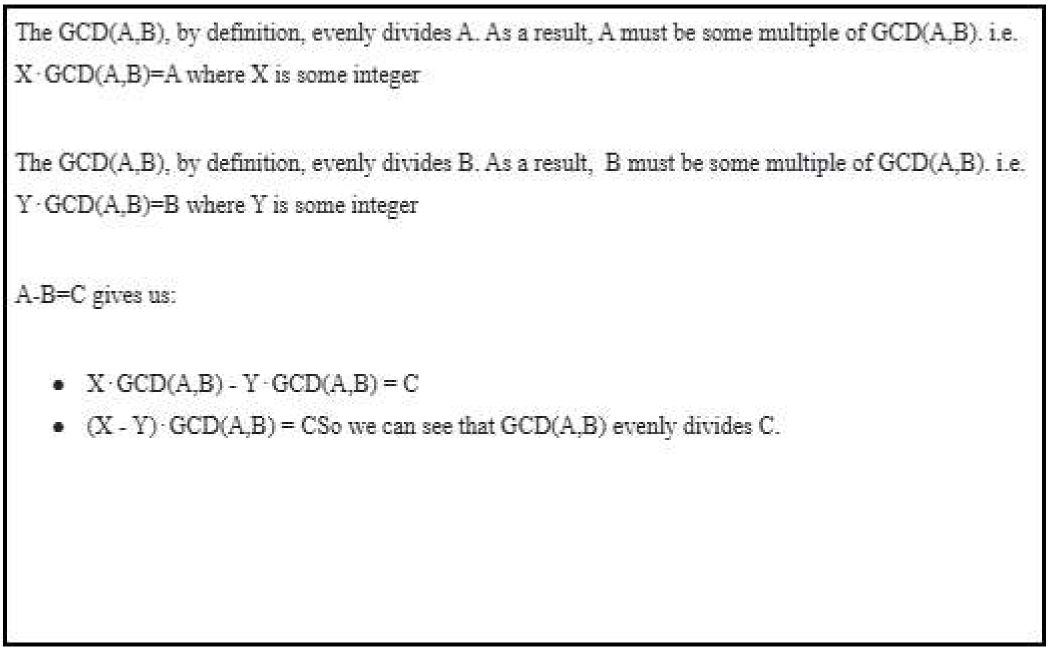
- The largest integer that can evenly divide A is A.
- All integers evenly divide 0, since for any integer, C, we can write $C \cdot 0 = 0$. So we can conclude that A must evenly divide 0.
- The greatest number that divides both A and 0 is A.

The proof for GCD(0,B)=B is similar. (Same proof, but we replace A with B).

To prove that GCD(A,B)=GCD(B,R) we first need to show that GCD(A,B)=GCD(B,A-B).

Suppose we have three integers **A,B** and **C** such that **A-B=C**.

Proof that the GCD(A,B) evenly divides C



The GCD(A,B), by definition, evenly divides A. As a result, A must be some multiple of GCD(A,B). i.e. $X \cdot GCD(A,B)=A$ where X is some integer

The GCD(A,B), by definition, evenly divides B. As a result, B must be some multiple of GCD(A,B). i.e. $Y \cdot GCD(A,B)=B$ where Y is some integer

A-B=C gives us:

- $X \cdot GCD(A,B) - Y \cdot GCD(A,B) = C$
- $(X - Y) \cdot GCD(A,B) = C$So we can see that GCD(A,B) evenly divides C.