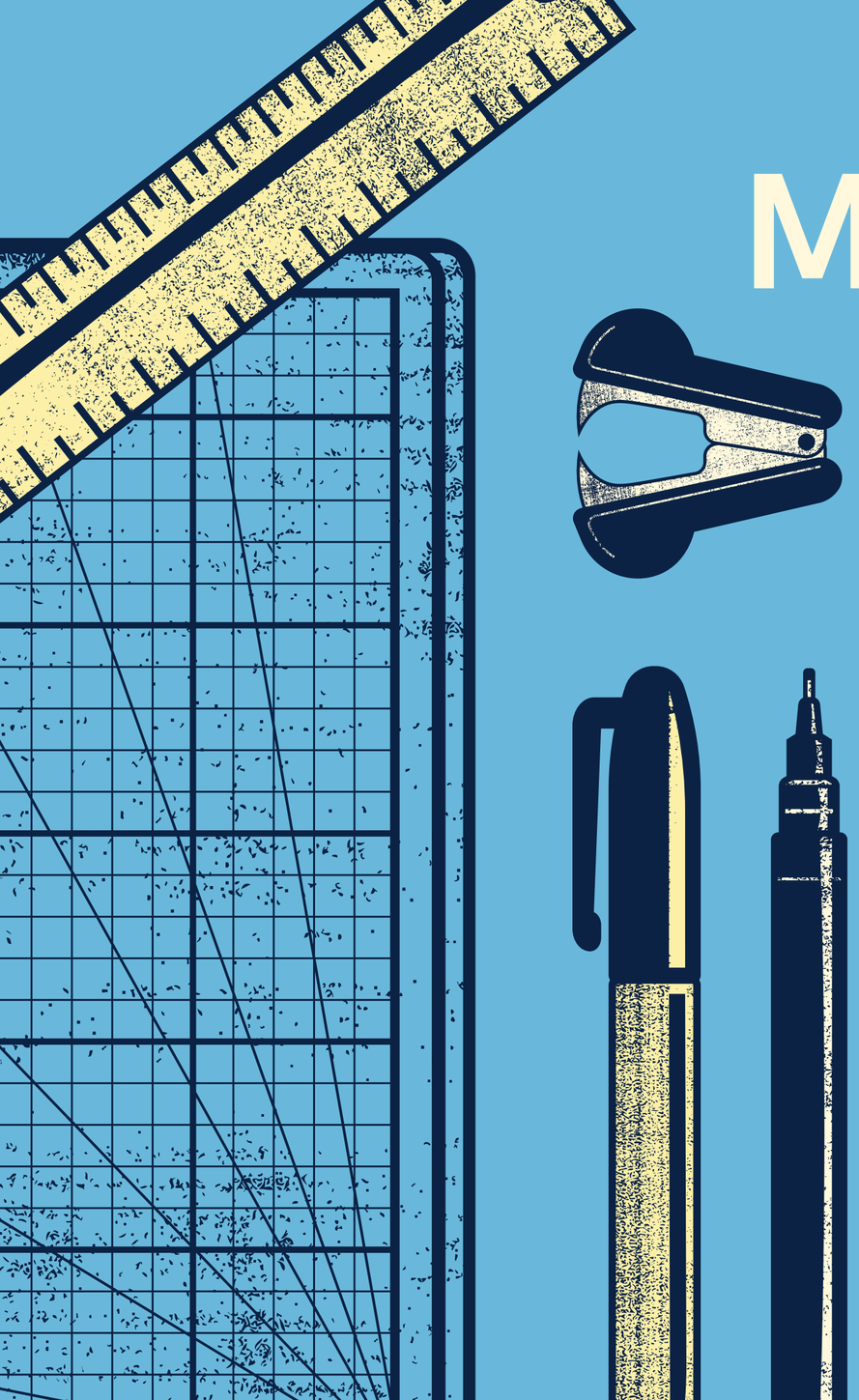


Mathematics Project

Title:Laws of logic

Abstract:

- Implementation of any propositions using laws of logic .

What is logic?

- A science that deals with the principles and criteria of validity of inference and demonstration
- In simple way a proper or reasonable way of thinking about something
- It helps us to understand reasoning behind issues which could be applied in other issues as well

Laws of logic

- Laws of logic are the Basic laws of Propositional Logic
- Fundamentally there are three laws of logic
    a. The law of contradiction : not(a and not a)
    b. The principle of identity : a is a
    c. The law of excluded middle : either “a” or not “a”

Problem Definition:

- For any propostion p&q using laws of logic evaluate
  a. $P \cup [P \cap (P \cup Q)]$
  b. $[(P \cup Q) \cap (P \cup \sim Q)] \cup Q$
  c. $(P \cup Q) \cap (\sim P)$
  d. $(P \cup Q) \cap \sim(\sim P \cap Q)$

TABLE 6 Logical Equivalences.

Equivalence	*Name*
$p \wedge \mathbf{T} \equiv p$ $p \vee \mathbf{F} \equiv p$	Identity laws
$p \vee \mathbf{T} \equiv \mathbf{T}$ $p \wedge \mathbf{F} \equiv \mathbf{F}$	Domination laws
$p \vee p \equiv p$ $p \wedge p \equiv p$	Idempotent laws
$\neg(\neg p) \equiv p$	Double negation law
$p \vee q \equiv q \vee p$ $p \wedge q \equiv q \wedge p$	Commutative laws
$(p \vee q) \vee r \equiv p \vee (q \vee r)$ $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$	Associative laws
$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$ $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	Distributive laws
$\neg(p \wedge q) \equiv \neg p \vee \neg q$ $\neg(p \vee q) \equiv \neg p \wedge \neg q$	De Morgan's laws
$p \vee (p \wedge q) \equiv p$ $p \wedge (p \vee q) \equiv p$	Absorption laws
$p \vee \neg p \equiv \mathbf{T}$ $p \wedge \neg p \equiv \mathbf{F}$	Negation laws

TABLE 7 Logical Equivalences Involving Conditional Statements.

$p \to q \equiv \neg p \vee q$

$p \to q \equiv \neg q \to \neg p$

$p \vee q \equiv \neg p \to q$

$p \wedge q \equiv \neg(p \to \neg q)$

$\neg(p \to q) \equiv p \wedge \neg q$

$(p \to q) \wedge (p \to r) \equiv p \to (q \wedge r)$

$(p \to r) \wedge (q \to r) \equiv (p \vee q) \to r$

$(p \to q) \vee (p \to r) \equiv p \to (q \vee r)$

$(p \to r) \vee (q \to r) \equiv (p \wedge q) \to r$

TABLE 8 Logical Equivalences Involving Biconditional Statements.

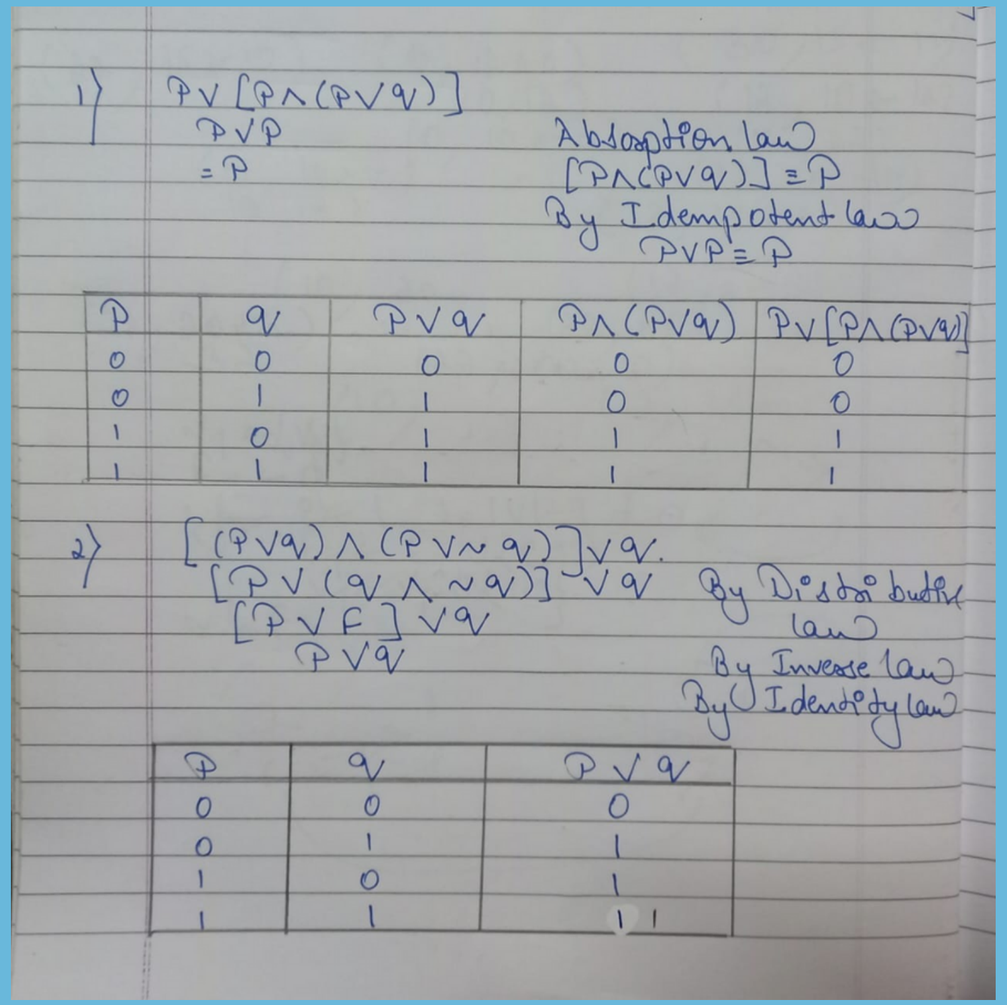
$p \leftrightarrow q \equiv (p \to q) \wedge (q \to p)$

$p \leftrightarrow q \equiv \neg p \leftrightarrow \neg q$

$p \leftrightarrow q \equiv (p \wedge q) \vee (\neg p \wedge \neg q)$

$\neg(p \leftrightarrow q) \equiv p \leftrightarrow \neg q$

Solutions:



1) $P \vee [P \wedge (P \vee q)]$

$P \vee P$

$= P$

Absorption law

$[P \wedge (P \vee q)] = P$

By Idempotent law

$P \vee P = P$

P	q	$P \vee q$	$P \wedge (P \vee q)$	$P \vee [P \wedge (P \vee q)]$
0	0	0	0	0
0	1	1	0	0
1	0	1	1	1
1	1	1	1	1

2) $[(P \vee q) \wedge (P \vee \sim q)] \vee q$.

$[P \vee (q \wedge \sim q)] \vee q$

$[P \vee F] \vee q$

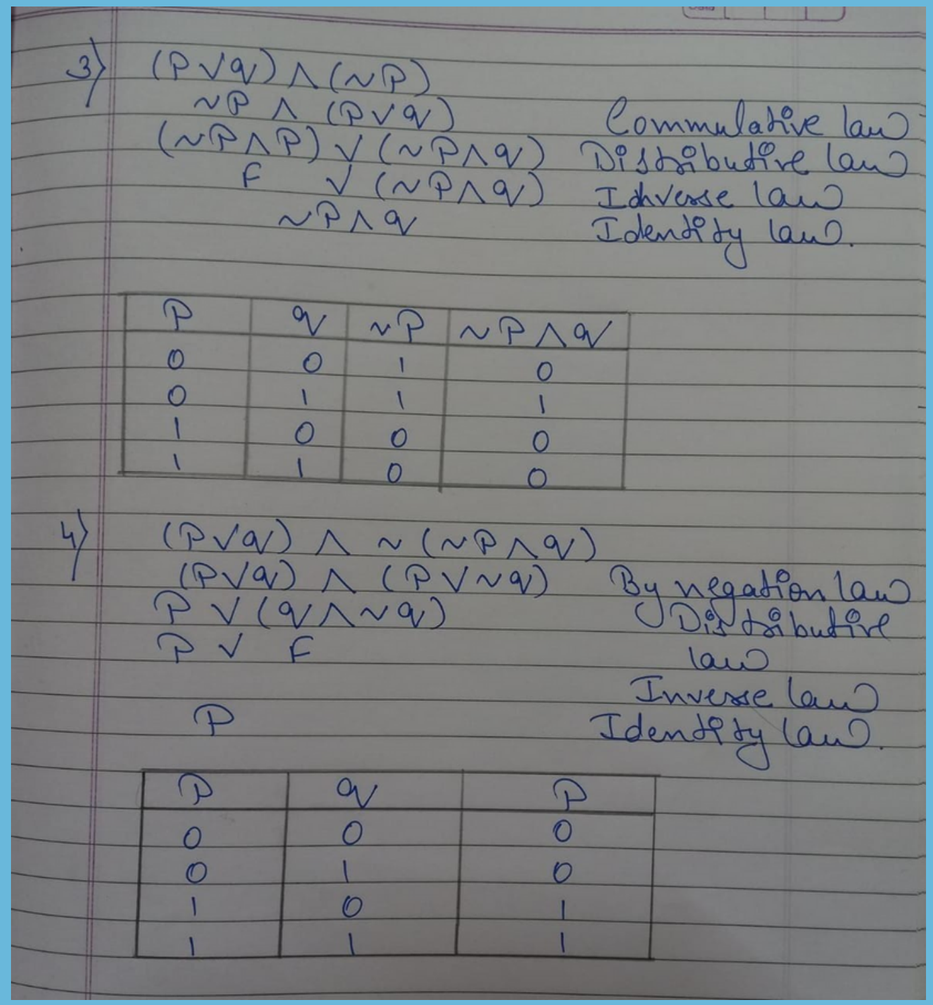
$P \vee q$

By Distributive law

By Inverse law

By Identity law

P	q	$P \vee q$
0	0	0
0	1	1
1	0	1
1	1	1



3) $(P \vee q) \wedge (\sim P)$

$\sim P \wedge (P \vee q)$ Commutative law

$(\sim P \wedge P) \vee (\sim P \wedge q)$ Distributive law

$F \vee (\sim P \wedge q)$ Inverse law

$\sim P \wedge q$ Identity law.

P	q	$\sim P$	$\sim P \wedge q$
0	0	1	0
0	1	1	1
1	0	0	0
1	1	0	0

4) $(P \vee q) \wedge \sim (\sim P \wedge q)$

$(P \vee q) \wedge (P \vee \sim q)$ By negation law

$P \vee (q \wedge \sim q)$ Distributive law

$P \vee F$ Inverse law

P Identity law.

P	q	P
0	0	0
0	1	0
1	0	1
1	1	1

Program Source Code to generate user input

```
def fun1():
    print("1.Full table 2.Particular value:")
    c=int(input())
    if(c==1):
        print("p\t q\t Result\n")
        for p in range(0,2):
            for q in range(0,2):
                r=p | (p & (p | q))
                print(p,end="\t")
                print(q,end="\t")
                print(r)
    else:
        print("Enter the value of p and q")
        p=int(input())
        q=int(input())
        r=(p | (p & (p | q)))
        print("Result:",end="\t")
        print(r)
```

```
def fun2():
    print("1.Full table 2.Particular value:")
    c=int(input())
    if(c==1):
        print("p\t q\t Result\n")
        for p in range(0,2):
            for q in range(0,2):
                r=((p | q) & (p | ~q)) | q
                print(p,end="\t")
                print(q,end="\t")
                print(r)
    else:
        print("Enter the value of p and q")
        p=int(input())
        q=int(input())
        r=((p | q) & (p | ~q)) | q
        print("Result:", end="\t")
        print(r)
```

```
def fun3():
    print("1.Full table 2.Particular value:")
    c=int(input())
    if(c==1):
        print("p\t q\t Result\n")
        for p in range(0,2):
            for q in range(0,2):
                r=((p | q) & (~p))
                print(p,end="\t")
                print(q,end="\t")
                print(r)
    else:
        print("Enter the value of p and q")
        p=int(input())
        q=int(input())
        r=((p | q) & (~p))
        print("Result:", end="\t")
        print(r)
```

```
def fun4():
    print("1.Full table 2.Particular value:")
    c=int(input())
    if(c==1):
        print("p\t q\t Result\n")
        for p in range(0,2):
            for q in range(0,2):
                r=(p | q) & ~(~p & q)
                print(p,end="\t")
                print(q,end="\t")
                print(r)
    else:
        print("Enter the value of p and q")
        p=int(input())
        q=int(input())
        r=(p | q) & ~(~p & q)
        print("Result:", end="\t")
        print(r)
```

```
def main():
    while True:
        print("Enter 1.Function1  2.Function2  3.Function3  4.Function4  5.Exit:")
        c = int(input("Enter your choice:"))
        if c == 1:
            fun1()
        elif c == 2:
            fun2()
        elif c == 3:
            fun3()
        elif c == 4:
            fun4()
        else:
            break

if __name__=='__main__':
    main()
```

Output:

```
"C:\Users\Sanket Patil\New folder\python.exe" "C:/Users/Sanket Patil/PycharmProjects/pythonProje/Maths.py"
Enter 1.Function1  2.Function2  3.Function3  4.Function4  5.Exit:
Enter your choice:1
1.Full table 2.Particular value:
1
p    q   Result

0   0   0
0   1   0
1   0   1
1   1   1
Enter 1.Function1  2.Function2  3.Function3  4.Function4  5.Exit:
Enter your choice:2
1.Full table 2.Particular value:
1
p    q   Result

0   0   0
0   1   1
1   0   1
1   1   1
```

```
Enter 1.Function1  2.Function2  3.Function3  4.Function4  5.Exit:
Enter your choice:3
1.Full table 2.Particular value:
1
p    q   Result

0   0   0
0   1   1
1   0   0
1   1   0
Enter 1.Function1  2.Function2  3.Function3  4.Function4  5.Exit:
Enter your choice:4
1.Full table 2.Particular value:
1
p    q   Result

0   0   0
0   1   0
1   0   1
1   1   1
```

```
Enter 1.Function1  2.Function2  3.Function3  4.Function4  5.Exit:
Enter your choice:1
1.Full table 2.Particular value:
2
Enter the value of p and q
1
1
Result: 1
Enter 1.Function1  2.Function2  3.Function3  4.Function4  5.Exit:
Enter your choice:2
1.Full table 2.Particular value:
2
Enter the value of p and q
1
0
Result: 1
Enter 1.Function1  2.Function2  3.Function3  4.Function4  5.Exit:
Enter your choice:5

Process finished with exit code 0
```

THANK U

Presented By:
Shambhavi Shirgurkar
Sanat K S
Sanket Patil
Sahana Tavari