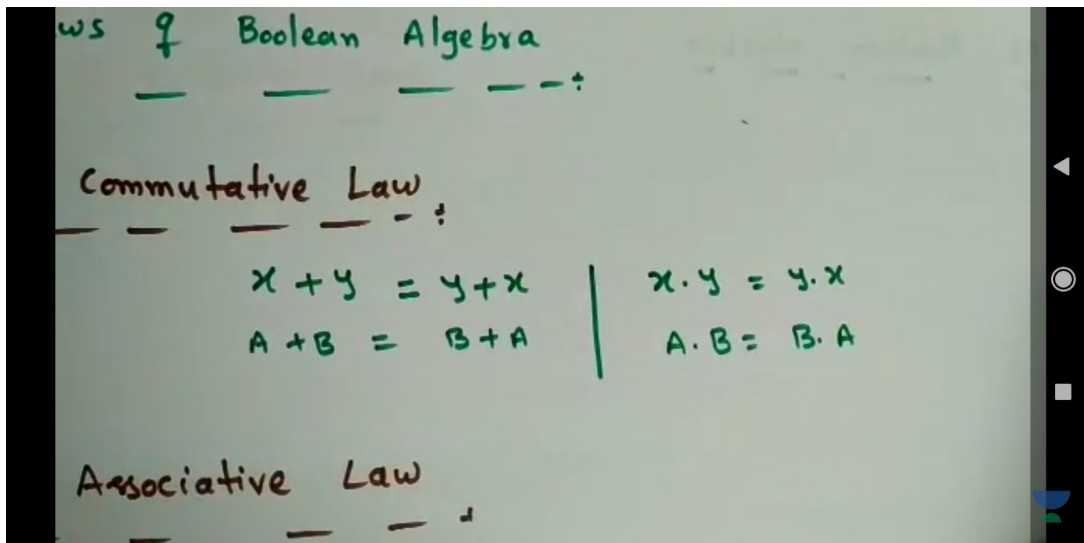
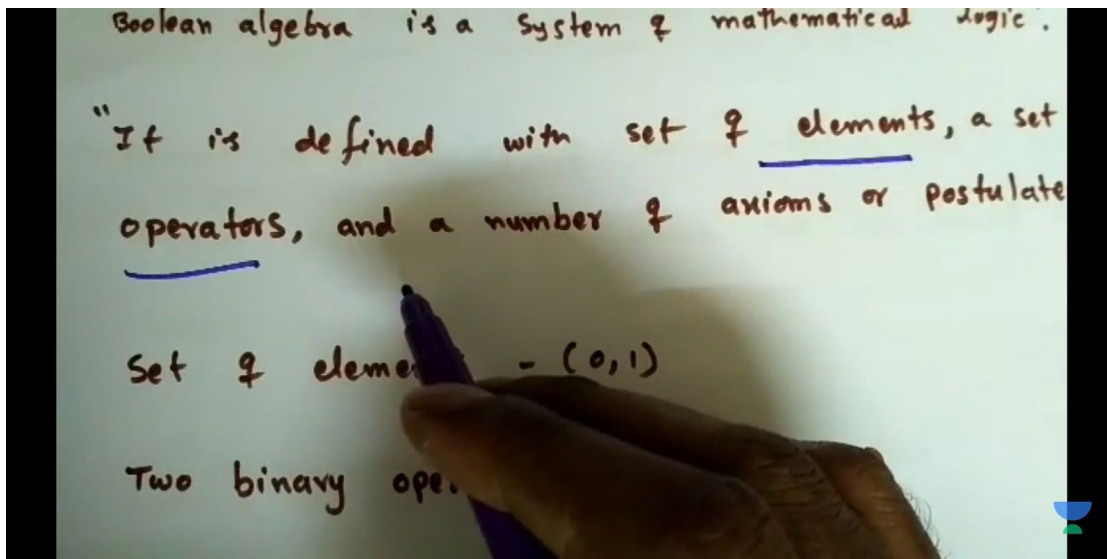


## DAILY ASSESSMENT FORMAT

Date:	28-05-2020	Name:	MOUNITHA D M
Course:	LOGIC DESIGN	USN:	4AL17EC055
Topic:	Boolean equation for digital circuits Combinational circuits Design of 7 segments decoder	Semester & Section:	6 <sup>TH</sup> SEM "A" SEC
Github Repository:	Mounitha_-ec055		

### FORENOON SESSION DETAILS

Image of session



# ws 9 Boolean Algebra

## Commutative Law

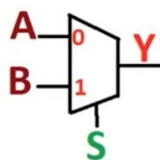
$$x + y = y + x$$

$$A + B = B + A$$

$$x \cdot y = y \cdot x$$

$$A \cdot B = B \cdot A$$

## Associative Law



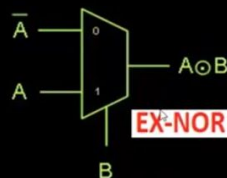
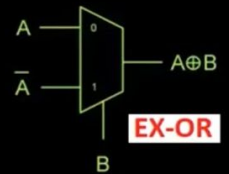
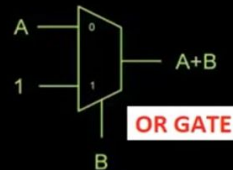
SELECTION (S)	OUTPUT(Y)
0	A
1	B

$$Y = A\bar{S} + BS$$

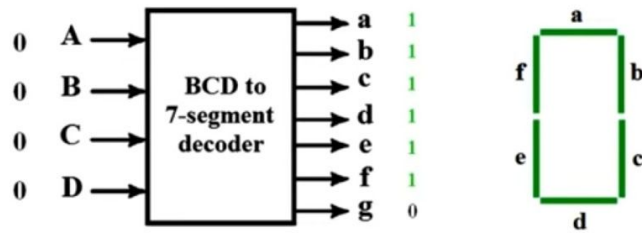
$$2^n - 1$$

$$2^n = \text{inputs}$$

$n$  = selection lines



## BCD to 7-segment decoder



A	B	C	D	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1

## Logic Design

Day - 1

Boolean Equations for digital circuits Combinational circuits : Conversion of Mux and Decoders to logic gates

Boolean Algebra or logic gate

Boolean Algebra '0' or '1'  
→ cost of the circuit  
→ Simple realization of a circuit

In 1854 George Boole developed an algebraic Boolean algebra is a system of mathematical logic.

Set of element - (0, 1)

Two binary operations - OR (+) and AND (.)

"AND operation" "OR operation"

$$\begin{array}{ll} \rightarrow 0 \cdot 0 = 0 & \rightarrow 0 + 0 = 0 \\ \rightarrow 0 \cdot 1 = 0 & \rightarrow 0 + 1 = 1 \\ \rightarrow 1 \cdot 0 = 0 & \rightarrow 1 + 0 = 1 \\ \rightarrow 1 \cdot 1 = 1 & \rightarrow 1 + 1 = 1 \end{array}$$

→ In Boolean algebra

$$\begin{array}{ll} A + A = A & A \cdot A = A \\ 1 + 1 = 1 & 1 \cdot 1 = 1 \end{array}$$

→ In ordinary algebra

$$\begin{array}{ll} A + A = 2A & A \cdot A = A^2 \\ 1 + 1 = 2 & 1 \cdot 1 = 1 \end{array}$$

→ In Binary number system

$$\begin{array}{ll} \rightarrow x + 0 = x & x \cdot 0 = 0 \\ \rightarrow x + 1 = 1 & x \cdot 1 = x \\ \rightarrow x + x = x & x \cdot x = x \\ \rightarrow x + \bar{x} = 1 & x \cdot \bar{x} = 0 \end{array} \quad (\bar{\bar{x}}) \text{ or } (x')' = x$$

Identity Element

OR operation

$$\begin{array}{l} x + 0 = x \\ 0 + x = x \end{array}$$

AND operation

$$x \cdot 1 = x = 1 \cdot x$$

The additive identity = '0'

The Multiplicative identity = '1'

## Laws of Boolean Algebra

### (1) Commutative Law

$$x + y = y + x$$
$$A + B = B + A$$

$$x \cdot y = y \cdot x$$
$$A \cdot B = B \cdot A$$

### (2) Associative Law

$$x + (y + z) = (x + y) + z$$

$$A + (B + C) = (A + B) + C$$

$$(1) \quad x(y + z) = xy + xz$$

$$A(B + C) = AB + AC$$

$$(2) \quad x + yz = (x + y)(x + z)$$
$$= x \cdot x + xz + xy + yz$$
$$= x + xz + xy + yz \Rightarrow x[1 + z + y] + yz$$
$$\Rightarrow x + yz$$

## Theorems of Boolean algebra

### (1) Absorption Theorem

$$x + xy = x$$

$$\rightarrow x(1 + y) \rightarrow x \cdot 1 \rightarrow x$$

$$A + AB = A$$

$$(b) \quad x + \bar{x}y = x + y$$

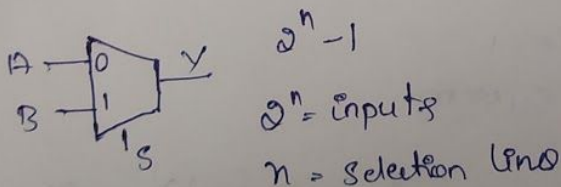
$$(x + \bar{x})(x + y)$$

## MUX to Logic gates Conversion

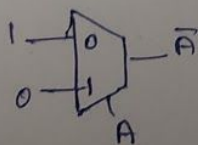
1) NAND, NOR universal gates

2) "Universal Logic"

3) MUX and Decoder are called "Universal Logic"



Inverter Design

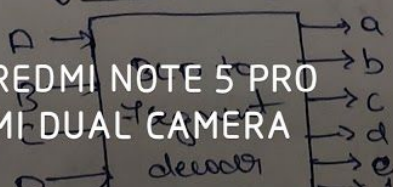


AND gate

$$Y = 0 \cdot \bar{B} + A \cdot B$$

$$Y = A \cdot B$$

Design of 4 segment decoder with common anode display



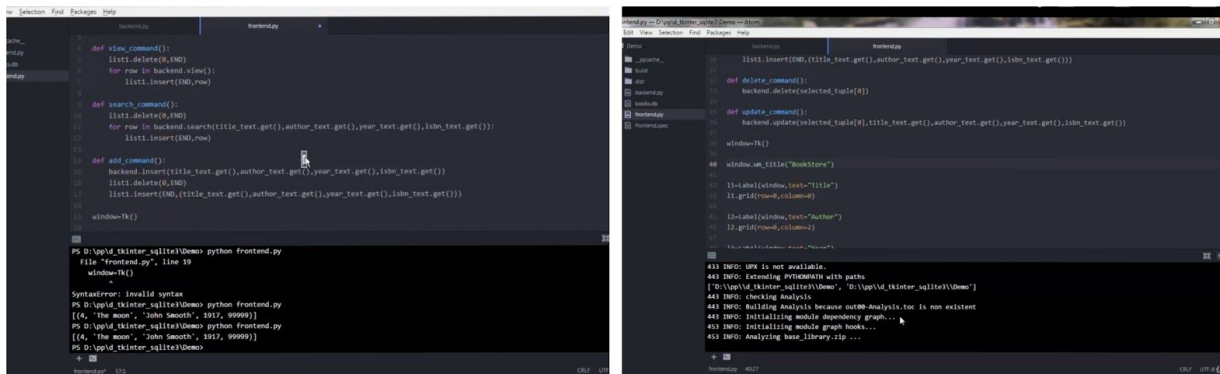
REDMI NOTE 5 PRO  
MIDUAL CAMERA



DATE	28-05-2020	Name:	MOUNITHA DM
Course:	PYTHON	USN:	4AL17EC055
Topic:	Build a Desktop Database Application	Semester & Section:	6 <sup>TH</sup> SEM "A" SEC

## AFTERNOON SESSION DETAILS

### Image of session



### ectures More

#### Section 23 - Application 5: Build a Desktop Database Ap...

- 80 ☒ Desktop Database App - How The ...  
Video - 02:25 mins - Resources (1)
- 81 ☒ User Interface Design  
Video - 05:54 mins
- 82 ☒ Frontend Interface  
Video - 13:28 mins
- 83 ☒ Backend  
Video - 24:28 mins
- 84 ☒ Connecting the Frontend to the Ba...  
Video - 17:31 mins
- 85 ☒ Connecting the Frontend to the Back...  
Video - 21:59 mins
- 86 Fixing the Bug (Practice)

### Lectures More

- 181 ☒ User Interface Design  
Video - 05:54 mins
- 182 ☒ Frontend Interface  
Video - 13:28 mins
- 183 ☒ Backend  
Video - 24:28 mins
- 184 ☒ Connecting the Frontend to the Ba...  
Video - 17:31 mins
- 185 ☒ Connecting the Frontend to the Ba...  
Video - 21:59 mins
- 186 Fixing the Bug (Practice)  
Article
- 187 Solution  
Article
- 188 ☒ Creating a Standalone Executable V...  
Video - 05:00 mins

Report – Report can be typed or hand written for up to two pages.

Day 9 - python.

28/05/2020

Application 5: Build a Desktop Database Application.

User Interface Design

A program that store the book information:

Title, Author

Year, ISBN

User can:

View all records

Search an entry

Add Entry

Update Entry

Delete

Close.

Frontend Interface

```
from tkinter import
window = Tk()
l1 = Label(window, text = "Title")
l1 = grid(row=0, column=0)
l2 = Label(window, text = "Author")
l2 = grid(row=0, column=1)
l3 = Label(window, text = "Year")
l3 = grid(row=1, column=0)
e1 = Entry(window, textvariable=)
e1 = grid(row=0, column=1)
author_text = StringVar()
e2 = Entry(window, textvariable=author_text)
e2 = grid(row=0, column=3)
b1 = Button(window, text = "View all", width=12)
b1 = grid(row=2, column=3)
b2 = Button(window, text = "Search Entry", width=)
b2 = grid(row=3, column=3)
```

### Backend

```
import sqlite3
```

```
def connect():
```

```
    conn = sqlite3.connect("books.db")
```

```
    cur = conn.cursor()
```

```
    cur.execute("CREATE TABLE IF NOT EXISTS book
```

```
        (id INTEGER PRIMARY KEY, title text, author text,
         Year integer, isbn integer)")
```

```
    conn.commit()
```

```
    conn.close()
```

### Connecting the frontend to the Backend

```
from tkinter import *
```

```
import backend
```

```
def view_command():
```

```
    row = backend.view()
```

```
    list1.insert(END, row)
```

```
Window = Tk()
```

```
l1 = Label(Window, text="Title")
```

```
l1.grid(row=0, column=0)
```

```
l2 = Label(Window, text="Author")
```

```
l2.grid(row=0, column=2)
```

```
l3 = Label(Window, text="Year")
```

```
l3.grid(row=1, column=0)
```

```
import backend
```

```
def view_command():
```

```
    list1.delete(0, END)
```

```
    row = backend.view()
```

```
    list1.insert(END, row)
```

```
def Search_command():
```

```
    list1.delete(0, END)
```

```
    row = backend.Search(title_text.get(), author_text.get(),
```

```
    Year_text.get(), isbn_text.get())
```

```
    list1.insert(END, row)
```



