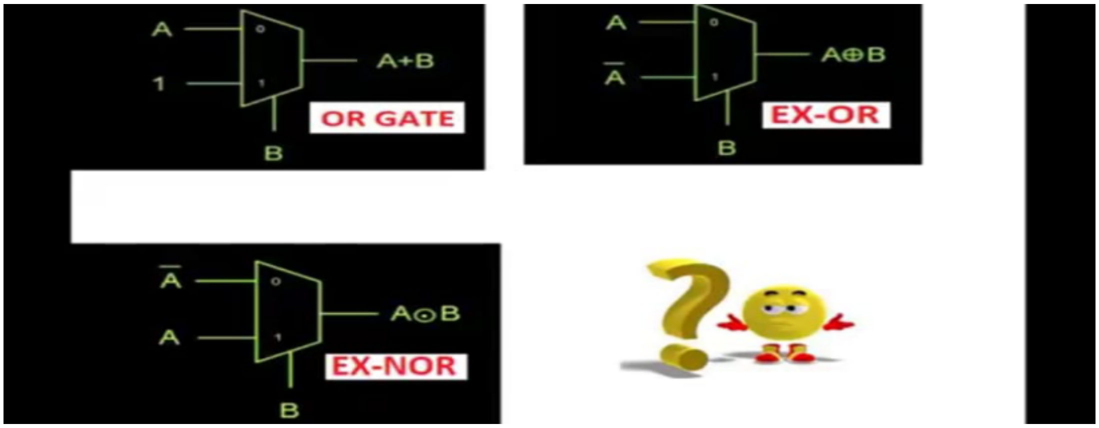
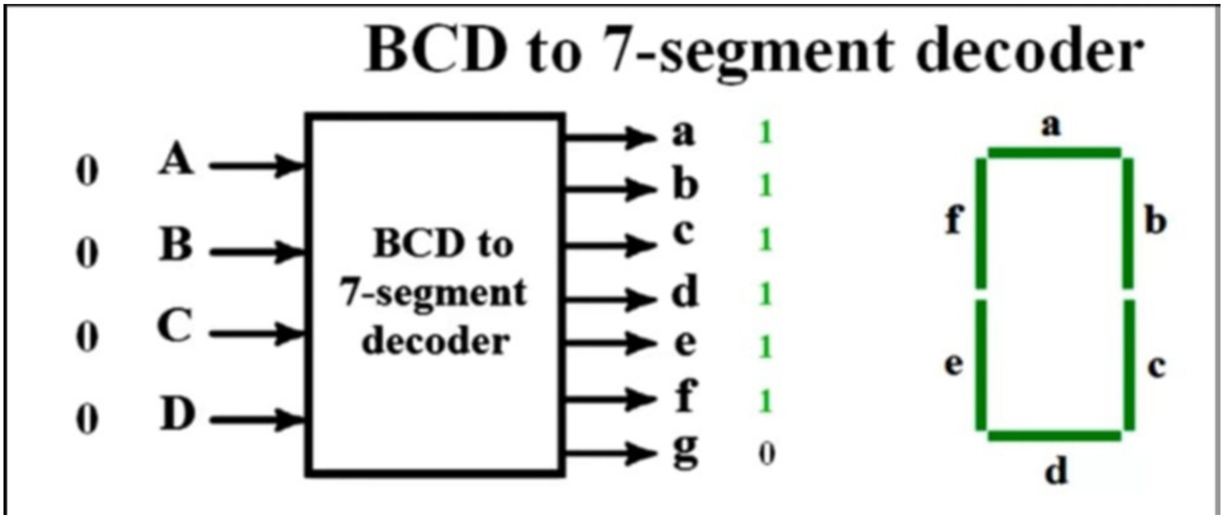
DAILY ASSESSMENT FORMAT

Date:	28/05/2020	Name:	Nichenametla Bhargavi
Course:	Logic Design	USN:	4AL17EC061
Topic:	Boolean Equations Combinational circuits BCD 7 segment Decoder	Semester & Section:	6th Sem A sec
Github Repository:	alvas-education-foundation/Bhargavi_Nichenametla		

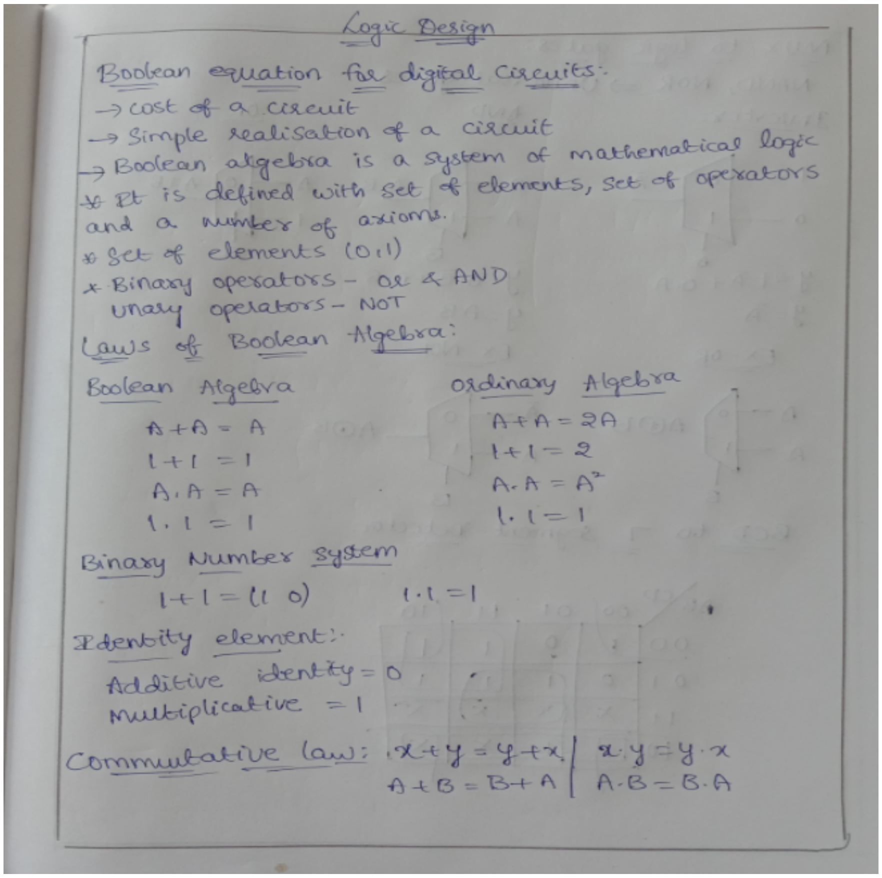
FORENOON SESSION DETAILS

Image of session





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



Logic Design

Boolean equation for digital Circuits:

→ cost of a circuit

→ Simple realisation of a circuit

→ Boolean algebra is a system of mathematical logic

* It is defined with set of elements, set of operators and a number of axioms.

* Set of elements (0,1)

* Binary operators – OR & AND
  Unary operators – NOT

Laws of Boolean Algebra:

Boolean Algebra	Ordinary Algebra
$A+A = A$	$A+A = 2A$
$1+1 = 1$	$1+1 = 2$
$A.A = A$	$A.A = A^2$
$1.1 = 1$	$1.1 = 1$

Binary Number system

$1+1 = (1\ 0)$ $\quad 1.1 = 1$

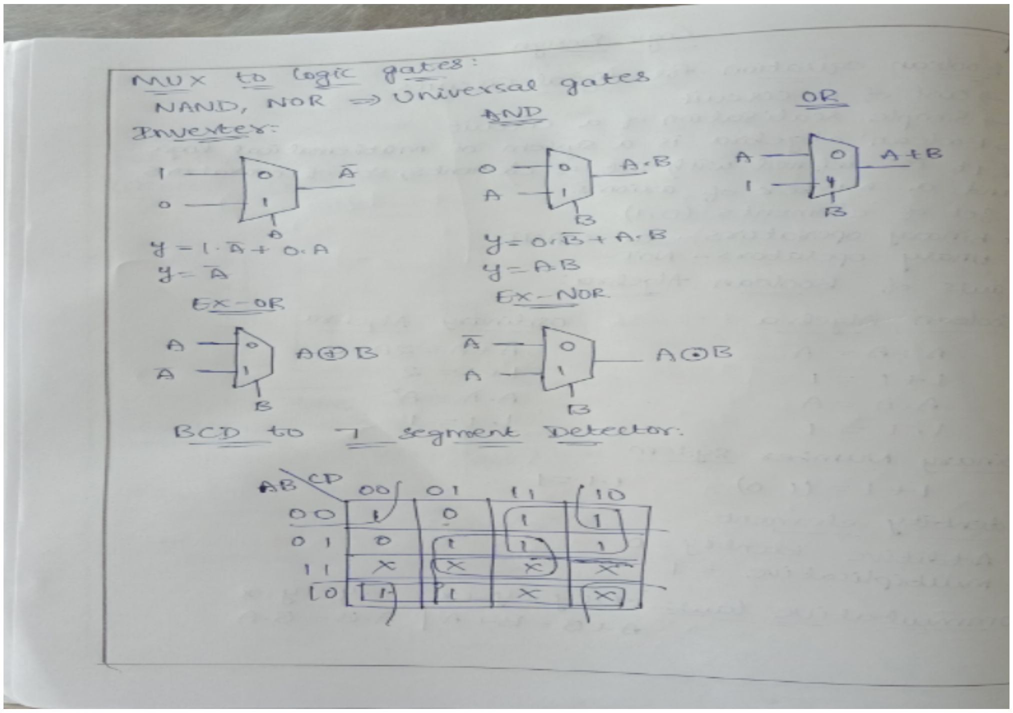
Identity element:

Additive identity = 0

Multiplicative = 1

Commutative law: $x+y = y+x$ | $x.y = y.x$

$A+B = B+A$ | $A.B = B.A$



MUX to logic gates:

NAND, NOR ⇒ Universal gates

Inverter:

1
0
0
1
$\bar{A}$
A

$y = 1.\bar{A} + 0.A$

$y = \bar{A}$

AND

0
A
0
1
A.B
B

$y = 0.\bar{B} + A.B$

$y = A.B$

OR

A
1
0
1
A+B
B

EX-OR

A
$\bar{A}$
0
1
$A \oplus B$
B

EX-NOR

$\bar{A}$
A
0
1
$A \odot B$
B

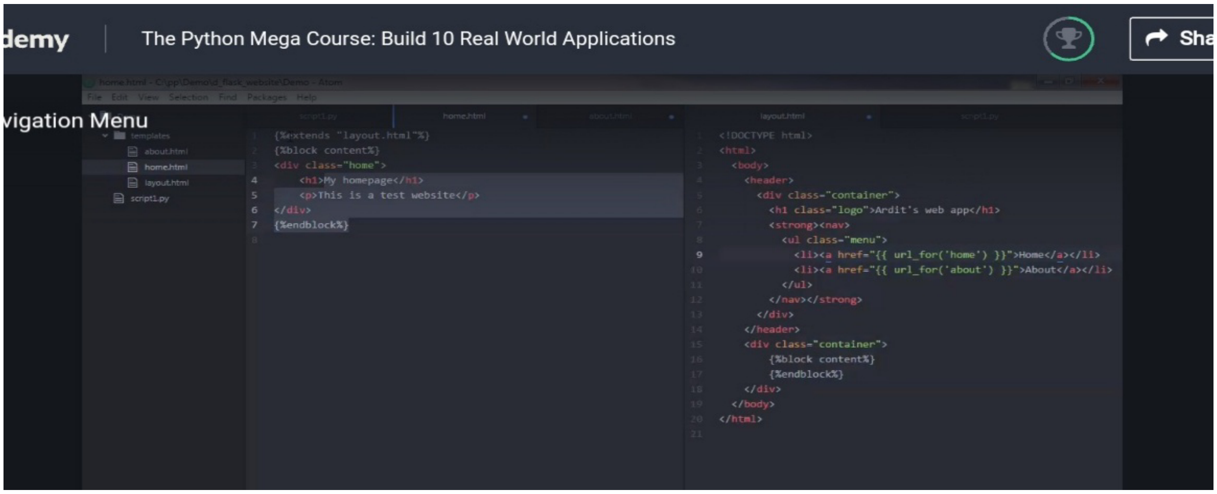
BCD to 7 segment Detector:

AB \ CD	00	01	11	10
00	1	0	1	1
01	0	1	1	1
11	X	X	X	X
10	1	1	X	X

Date:	28/05/2020	Name:	Nichenametla Bhargavi
Course:	Python	USN:	4AL17EC061
Topic:	Application 4: Build a Personal Website with Python and Flask	Semester & Section:	6th Sem A sec

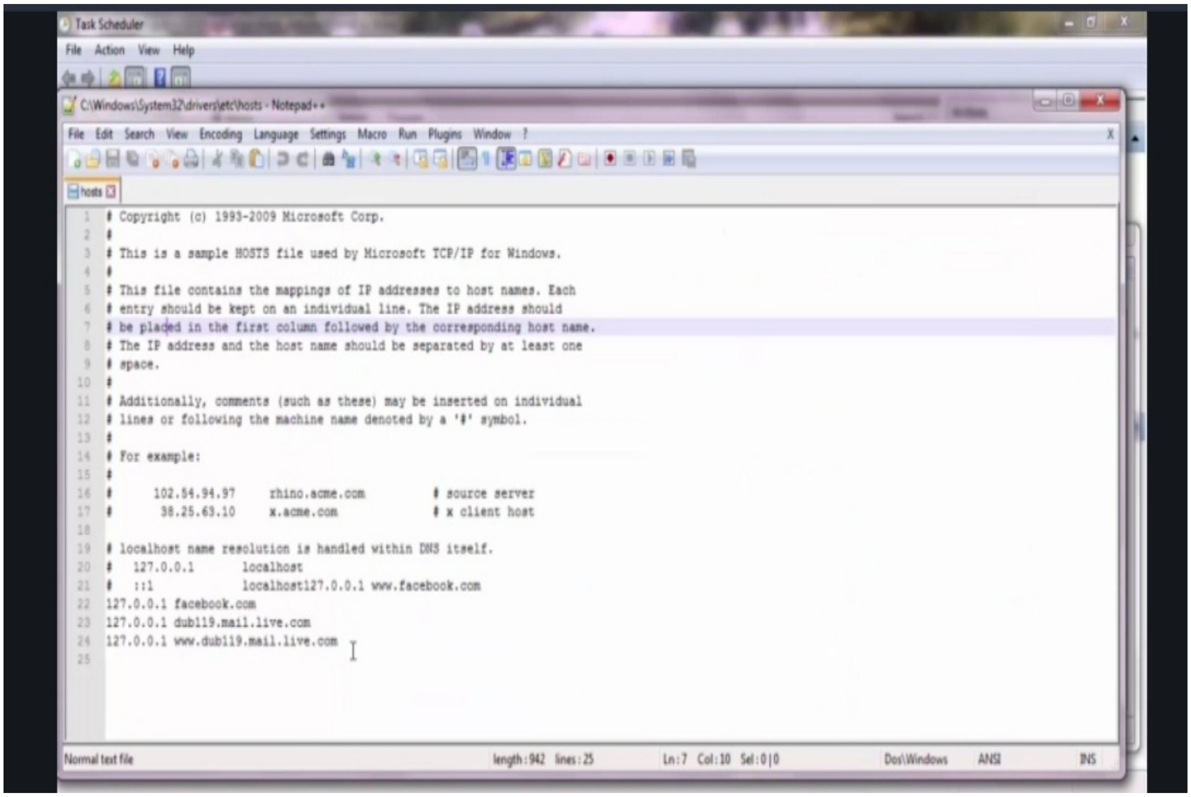
AFTERNOON SESSION DETAILS

Image of session


demy
The Python Mega Course: Build 10 Real World Applications
Sha
home.html - C:\pp\Demo\d_flask_website\Demo - Atom
File Edit View Selection Find Packages Help
vigation Menu
templates
about.html
home.html
layout.html
script1.py
script1.py
home.html
about.html
layout.html
script1.py

```
{%extends "layout.html"%}
{%block content%}
<div class="home">
    <h1>My homepage</h1>
    <p>This is a test website</p>
</div>
{%endblock%}
```

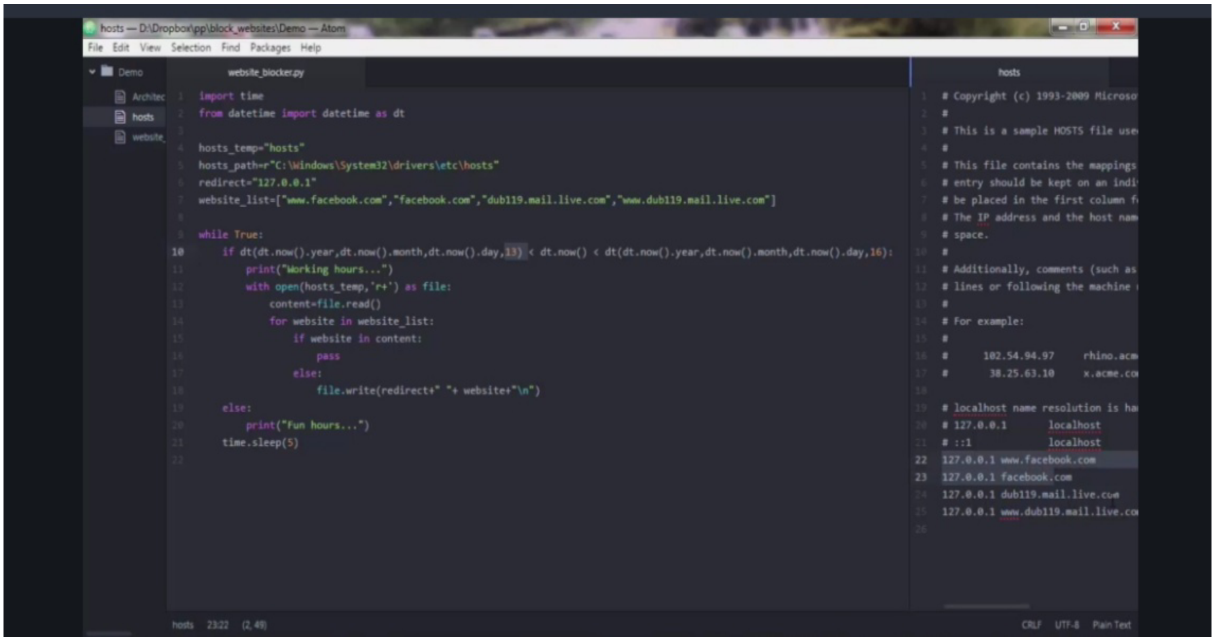
```
<!DOCTYPE html>
<html>
  <body>
    <header>
      <div class="container">
        <h1 class="logo">Ardit's web app</h1>
        <strong><nav>
          <ul class="menu">
            <li><a href="{{ url_for('home') }}">Home</a></li>
            <li><a href="{{ url_for('about') }}">About</a></li>
          </ul>
        </nav></strong>
      </div>
    </header>
    <div class="container">
        {%block content%}
        {%endblock%}
    </div>
  </body>
</html>
```


Task Scheduler
File Action View Help
C:\Windows\System32\drivers\etc\hosts - Notepad++
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
hosts
Normal text file
length : 942 lines : 25
Ln : 7 Col : 10 Sel : 0 | 0
Dos\Windows
ANSI
INS

```
# Copyright (c) 1993-2009 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#      102.54.94.97     rhino.acme.com          # source server
#       38.25.63.10     x.acme.com              # x client host

# localhost name resolution is handled within DNS itself.
#   127.0.0.1       localhost
#   ::1             localhost127.0.0.1 www.facebook.com
127.0.0.1 facebook.com
127.0.0.1 dub119.mail.live.com
127.0.0.1 www.dub119.mail.live.com
```

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


hosts — D:\Dropbox\pp\block_websites\Demo — Atom
File Edit View Selection Find Packages Help
Demo
Architec
hosts
website_
website_blocker.py
hosts
hosts 23:22 (2, 49)
CRLF UTF-8 Plain Text

```python
import time
from datetime import datetime as dt

hosts_temp="hosts"
hosts_path=r"C:\Windows\System32\drivers\etc\hosts"
redirect="127.0.0.1"
website_list=["www.facebook.com","facebook.com","dub119.mail.live.com","www.dub119.mail.live.com"]

while True:
    if dt(dt.now().year,dt.now().month,dt.now().day,13) < dt.now() < dt(dt.now().year,dt.now().month,dt.now().day,16):
        print("Working hours...")
        with open(hosts_temp,'r+') as file:
            content=file.read()
            for website in website_list:
                if website in content:
                    pass
                else:
                    file.write(redirect+" "+ website+"\n")
    else:
        print("Fun hours...")
    time.sleep(5)
```

```
# Copyright (c) 1993-2009 Microso
#
# This is a sample HOSTS file use
#
# This file contains the mappings
# entry should be kept on an indi
# be placed in the first column f
# The IP address and the host nam
# space.
#
# Additionally, comments (such as
# lines or following the machine
#
# For example:
#
#      102.54.94.97     rhino.acm
#       38.25.63.10     x.acme.co

# localhost name resolution is ha
# 127.0.0.1       localhost
# ::1             localhost
127.0.0.1 www.facebook.com
127.0.0.1 facebook.com
127.0.0.1 dub119.mail.live.com
127.0.0.1 www.dub119.mail.live.co
```

Hi! In case you didn't understand the `if not any(website in line for website in website_list)` part in the previous video, here is another example:

```
>>> lines   = ["trees are good", "pool is fresh", "face is roun
>>> website_list = ["face", "clock", "trend"]
>>> for line in lines:
...     any(website in line for website in website_list)
...
False
False
True
```

We start iterating over the items of `website_list` using a `for` loop. In the first iteration we would have:

`any(website in "trees are good" for website in website_list)`

Inside the parenthesis of `any()` there's another loop that iterates over `website_list`:

```
("face" in "trees are good")
("clock" in "trees are good")
("trend" in "trees are good")
```

If `any` of the above is `True` you get the expression evaluated to True. In this case none of them is True, so you get `False`.

If you want to return `True` (if all of them are `True`), use `all()` instead of `any()`.

`all()` instead of `any()`.

So, the part `any(website in line for website in website_list)` will either be equal to `True` or `False`.