| Sr. No. | Topic | Date | Remarks | Signature |
|---|---|---|---|---|
| 1A] | write a python prg to implement BFS | 17/6/25 | | |
| 1B] | write a python prg to implement DFS | 24/6/25 | | |
| 2A] | write a python prg to implement Tower of Hanoi | 1/7/25 | | |
| 2 B] | write a python prg to solve water jug problem | 8/7/25 | | |
| 3] | Simulation of Tic Toe in python | 15/7/25 | | |
| 4] | write a python prg to solve missionaries and cannibal problem | 24/7/25 | | |
| 5A] | Derive the assosiative law | 5/8/25 | | |
| 5B] | Derive the distributive law | 12/8/25 | | |
| 6] | Write python prg for N queen problem | 28/8/25 | | |
| 7] | Implement hill climbing algorithm | 2/9/2025 | | |
| 8] | Implement Travelling Salesman (solo traveller) algorithm. | 9/9/25 | | |

Practical 1:

Starting Rasphian OS, familiarising with Raspbeery pi components and interface, connecting to ethernet, Monitor, USB.

-> What is Raspbeery Pi?

The Raspbeery pi is a mini computer that was specifically created to make tech learning easier. It has lot of components for computer-based projects, like USB ports, an ethernet port, an SD card slot Wi-Fi antenna ports, & more

It does not come with peripherals, like cables, a keyboard, a mouse, or a monitor. It is great for learning prg languages, like python, Scratch, and Wolfrom. Most Raspberry Pi enthusiasts like making single-process builds to show off their do-it-yourself talents.

For example, you could create a ~~direct~~ dedicated gaming device or an external storage box for movies & music. There are a plethora of Raspberry Pi projects that cover all manner of possibilities, each one with diffrent specification. We have a guide for getting started with Raspberry Pi to help one with diffrent specifications. We have ~~g~~ a guide for getting started with Raspberry Pi to help you understand what you will need for your first (or next) of project

What you will need

The Raspberry Pi ships as just the single-board minicomputer. There are a few additional components you will need before you can get starded. so, when making your purchase, Keep in mind that you'll need the following extras.

1) Raspberry Pi - There are six diffient models of Raspbeerypi The Pi 2 Model B or Pi 1 Model B+ and Pi 3 Model B are ideal for beginner projects because they are the most versatile and have the widest range of capabitities. The Pi 3 Model B has the added bonus of having a quadcore processor & 1GB of RAM so it supports heavier operating Systems, like Ubuntu and Microsoft 10. The model A+ is a powerful boards for building robotics, but doesn't have an Ethernet port and only comes with one USB port. So, it's better for people that are a little more savvy with engineering technology. Raspberry Pi zero is basically a miniature version of the Model A+, but has a more robust computing power. It has a micro USB port and mini HDMI port for 1080p output compatibility but doesn't have wireless capability It only costs $5 and Adafruit sells V1.3 for just $5, but you can only buy one per order. The Raspberry Pi zero W is the same single-board computer as the standard zero but does not support wireless & Bluetooth connectivity. It costs $10 on Adafruit, but you can order ~~of~~ one per day

2 Power supply - you will need a 5V micro-USB power supply You can find them for really cheap online. You may even have one from a non-apple mobile device lying around the house I recommend the Canakit 5V power supply

3 USB keyboard

4 USB mouse - If you prefer to use a Bluetooth keyboard and mouse, you could just get a Bluetooth adapter. I have a Kinivo BTD-400, but there & dozens of different brands out there.

5 microSD card - The microSD card must have a least 8GB of storage. You can purchase one that comes per-loaded with Raspberry Pi's New Out of Box Software (NOOBS), but you can also download the software for free from the website, so there is no need to purchase a special NOOBS microSD card.

Practical 2

Displaying different LED patterns with Raspberry Pi.

Components Required:

Here we are using Raspberry Pi 2 Model B with Raspbian OS. All the basic Hardware and Software requirements are previously discussed, you can look it up in the Raspberry Pi Introduction, other than that we need:

Connecting pins

220 Ω or 1K Ω resistor

LED

Bread Board

Circuit Explanation:

As shown in the circuit diagram we are going to connect an LED between PIN40 (GPIO21) and PIN39 (GROUND). As Said eariler, we cannot draw more than 15mA from any one these pins, so to limit ~~he~~ the current we are connecting a 220 Ω or 1K Ω resistor in series with the LED

Program 1:

```
import RPi.GPIO as IO
import time
 for i in range(0,25):
     IO.setmode(IO.BOARD)
as PIN40
    IO.setup(40.IO.OUT)
    IO.output(40.1)
    Time.sleep(0.5)
    IO.cleanup()
    time.Sleep(0.5)
```

Program 2:

```
import RPi.GPIO as GPIO
import time
x = 1
numTimes=int(input("Enter total Number Of times to
              blink:"))
Speed=float(input("Enter length of each blink(seconds):"))
GPIO.setwarmings(False)
GPIO.setmode(GPIO.BOARD)
GPIO.setup(40.GPIO.OUT)
def Blink (numTimes,speed):
      for i in range(0,numTimes):
          GPIO.output(40, True)
        print("Iteration",(i+1))
        time.sleep(speed)
        GPIO.output(40,False)
        time.sleep(speed)

Blink(numTimes,Speed)
print("Done")
```

No. : 07

Date :

Practical 7:

Capture image and videos wing libcamera

libcamera is a new Software libary aimed at supporting ~~ti~~ linux operating complex camera systems directly from the system. In the case of the Raspberry Pi it enables we to drive the camera System directly from open source code running on ARM Proressors

libcamera presents a C++ API to applications and works at the level of configuring the camera and than allowing an application to request image frames. Thee image buffers reside in system memory and can be passed directly to still image encodes (such as JPEG) or video encoders. (such as h.264), through such ancillary functions as encoding image or displaying them are strictly ~~to~~ beyond the purview of libcamera itself

libcamera-jpeg

libcamera-jpeg is a simple still image capture application It deliberately avoids some of the additional Feature of libcamera-still which attempts to emulate ~~rasp~~ raspistill more fully As such the code is Significantly easier to understand, and in practice Still provides many of the same features.

To capture a full resolution JPEG image use

```
libcamera-jpeg-o test.jpg
```

for more parameters, place use the --help parameter to see

```
libcamera-jpeg-help
```

## Practical-4

IOT based web controlled Home Automation using Raspberry Pi:

Required Components:-

For this project, the requirements will fall under two categories. Hardware and Software

Hardware Requirements:
1. Raspberry Pi 3 (Any other version will be nice)
2. Memory card 8 or 16GB running Raspbian Jessie
3. Jumper wires
4. LEDs to lest
5. Breadboard and Jumper cables
6. 220 or 100 ohms resistor

Softwar Requirements:
Asides the Raspbion Jessie operating System running on the raspberry Pi, we will also be wing the WebIOPi frame notepad ++ running on your PC and filezila to copy fies from the PC to the raspberry Pi, especially the web opp files.

Also you dont need to code in python for this Home Automation Project, WebIOPi will do all the work

Preparing the Raspberry Pi:
To update the raspberry Pi below commands & then the Pi

```
sudo apt-get update
Sudo apt-get upgrade
Sudo reboot
```

With the done the next thing is for us to install the webIOPi framework.
Make sure you are in home directory using:-

```
Cd
```

1.6 Name Your Bot: BotFather will ask for your bot. This name is what users will see:
1.7 Create a Username: You'll then be prompted to create a unique username for your bot. This username must end with the word bot (e.g. mytestbot)
1.8 Obtain access token /token

copy the token
/mybots

Step 2: setup Raspberry Pi

Note: if you already set the Pi then skip this section

Step 3: Install TelegramBot on Raspberry Pi
3.1 open Putty

3.2 connect Pi via SSH/VNC

enter pi ip address
pi username
we are in
3.3 Install "Python Package Index"

Step 4: Run the python code

4.1 Clone the git

```
git clone --recursive https://github.com/vysheng/tg.git
&& cd tg
```

```
git clone. https://github.com/salmanfarisvp/TelegramBot.git
```

4.2 paste your Bot Token here

```
bot= teleport.Bot('Bot Token')
python TeleLed.py
```

```
 bot.sendMessage(chat_id, off(11))

bot=telepot.Bot('565260611:AAGniNIOWPiEOck(ksOm6z
                 Mh5oS1DoEqkAI')
bot.message_loop(handle)
print('I am listening...')

while 1:
  try:
    time.Sleep(10)

 except KeyboardInterrupted')
  GPIO.cleanup()
  exit()

except:
     Print('Other error or exception occured!')
     GPIO.cleanup()
```

TeleCmd.py

```
import time, datetime
import telepot
From telepot.loop import MessageLoop
now =datetime.datetime.now()
def action (msg):
    chat_id = msg['chat']['id']
    command = msg['text']
    print('Received: %s' % command)
  if command == '/hi':
     telegram_bot.sendMessage(chat-id, str("Hi! DTSS
 BOT Here"))
elif command == '/time':
   telegram-bot.sendMessage(chat-id, Str(now.hour)+
Str(":") + str(now.minute))
elif command == '/logo':
     telegram-bot.send Photo (Chat-id, photo =
"http://pcsoftmumbai.com/pcs/wpcontent/uploads/2017
 05/logocopy.jpg")
```

```
elif command == '/file':
    telegram_bot.sendDocument (chat_id ,document=open('/home
/pi/Telecmd.py'))
elif command = ='/audio':
    telegram_bot.sendAution (chat_id,audio=open('/home/pi/
    piano.mp3','rb'))
elif command = = '/photo':
    telegram_bot.send Photo (chat_id,photo=open('/home/pi/
    photo.jpeg','rb'))
    telegram_bot = telepot.Bot('2013597168:AAHgjo7uffbLIQ
    WyGCufP35-TQVzcOacyNE')
print (telegram_bot.getMe()
MessageLoop(telegram_bot, action).run_as_thread()
print('up and Running....')

while 1:
        Time.sleep(10)
```

Step5: Connect LED to Pi
Step6: Send command

6.1 Start our Bot
6.2 Send "on"&"off"

Look at your Pi, you can see the LED on and off when you send "on" and "off" to our bot.
Connection between pi and LED

Code

TeleLed.py

```
import Sys
import time
import telepot
import RPi.GPIO as GPIO

def on(pin):
        GPIO.output(pin,GPIO,HIGH)
        return
def off(pin):
        GPIO.output(pin.GPIO.LOW)
        return
GPIO.Setmode (GPIO.BOARD)
GPIO.Setup (11.GPIO.OUT)

def handle(msg):
     chat_id=msg['chat']['id']
     command=msg['text']

print('Got command: %.s'%. command)

if command == 'on':
     bot.sendMessage(chat_id, on(11))
elif command == 'off':
```

Practical 5

Controlling Raspberry Pi with WhatsApp

Host a Telegram Bot on your Raspberry Pi and ~~Ch~~ chat with your brand new IoT devices!

Things used in this project

Hardware components

| | |
|---|---|
| Raspberry Pi 3 Model B | ×1 |
| LED (generic) | ×1 |
| Jumper wires (generic) | ×1 |
| Breadboard (generic) | ×1 |

Story

Learn how to use the Telegram Bot, host a Telegram Bot on your Raspberry Pi, and use the messaging app to interact with your device

Step 1: Open Telegram app in your System or mobile

1.1 Open Telegram app in your System or mobile

Download from here: Telegram

1.2 Start "BotFather"
1.3 Open "BotFather"
1.4 Start "BotFather"
~~1.5~~

1.5 Create a new Bot /start
/new bot
Bot: name332_bot

~~Use wget to get the file from their Sourceforge page: wget http://Sourceforge.net/projects/webIOPi-0.7.1.forge~~

Password is raspberry

This login can be removed later if desired but even your home automation System deserves Same extra level of security to prevent just anyone with the IP ~~coll~~ controlling appliances and IOT devices in your home.

After the login, look around and then click on the GPIO header link

For this test we will be connecting on LED to GPIO 17, So go on and set GPIO 17 as an output

with this done, connect the led to your raspberry pi as shown in the schematics below

After the connection, go back to the webpage & click the pin 11 button to turn on as off the LED. This way we can control the Raspberry Pi GPIO using webIO Pi

After the test, if everything coorked as decribed then we can go back to the terminal and Stop the program with CTRL + C. If you have any issue with this setyp then hit up via the ~~Fo~~ Comment section

Use wget to get the file from their source forge page:

```
weget http://source forge.net/projects/web IOPi - 0.7.1 for.gz
```

When download is done, extract the file and go into the directory:

```
for Wzf webIOPi - 0.7.1 tar.gz
cd webIOPi-0.7.1/
```

At this point before running the setup, we need to install a patch as this version of the webIOPi does not work with the raspberry pi 3 which I am using and I couldn't find a version of the webIOPi that works expressly with the pi 3

Below commands are used to install patch while still in the webIOPi directory

```
weget https://raw.githubusercontent.com/doublebind/raspi/moster/webiopi-pi2bplus.patch
patch-pi-i webiopi-pi2bplus.patch
```

Then we can run the setup installation for the webIOPi. using:

```
Sudo./setup.sh
```

Keep saying yes if asked to install any dependencies during Setup.installation when done, reboot your pi: sudo rebout

Test web IOPi Installation:

Befor jumping in to Schematies and codes With the raspberry Pi back on, we will need to test our webIOPi installation to be sure everything works fine as desired Run the command.

```
Sudo webiopi-d -c/etc/webiop./config
```

After issuing the command above on the pi, point the web browser of your computer connected to the raspberry pi to http://raspberry pi.mshome.net:8000 or http://thepi's IPaddress:8000

The System will prompt you for username & password Username is webiopi

```
libcamera-Still -t0 --autofocus-range macro
```

For OV5647/IMX219 and IMX477 Camera Module

libcamera-vid

libcamera-vid is the video Capture application. By default it wes the Raspbeerry Pi's hardware H.246 encoder. It will display a preview window & write the encoded bitstream to the Specified output. for example, to write a 10 second video to file use

```
libcamera-vid -t1000 -o test.h264
```

For cameras with focus motors, we have added autofocus parameter to enable single autofocus & continuous autofocus functions.

Single autofocus

```
libcamera-Still -t0 --autofocus-mode auto
```

Continuous automatic photo taking, 1 short per second within 5 seconds

```
libcamera-vid -t5000 --date time -n --timelapse 1000
```

No. : 08

Date :

libcamera - still

libcamera-still is very Similar to libcamera-~~&~~jpeg but supports more of the legacy raspistill options. As before, a single image can be captured with

```
libcamera-still -o test.jpg
```

Continow automatic photo taxing, 1 Shot per second with 5 seconds

```
libcamera-still -t5000 -datetime -n --timelapse 1000
```

For more parameters, please we the --help parameter to see ~~th~~ libcamera-still --help

CONTINUOUS AUTOFOCUS

```
libcamera-still -t0
```

```
libcamera-still -to --autofocus-mode continuous
```

SINGLE AUTOFOCUS

```
libcamera-still -t0 --autofocus-mode auto
```

ADJUST LENS POSITION

```
libcamera -still -t0 --autofocus-mode manual --lens-position 5
```

AUTOFOCUS BEFORE CAPTURE IMAGES

```
libcamera -still -t0 --autofocus-on-capture
```

SET FOCUS RANGE

```
libcamera-still -t0 --autofocus-rangenormal
```

7 A monitor or TV that Supports HDMI or composite Video - You can use an older composite video display, but HDMI works better and Supports audio transfers.

8. An HDMI cable or composite video cable, depending on what the screen you use supports

~~9 An HDMI cable or composite video cable, depen~~

9 An ethernet cable (or Wi-Fi dongle) - A connection to the Internet is not required for setup, but many Raspberry Pi projects use them.