OPEN ENDED EXPERIMENT **Date: 12 April 2023**

Aim:

(a) System Resource Monitor: Identify current available and utilized resources of the system e.g., CPU, memory, I/O, and bandwidth. Your program should be able to log historical system resources and capable to show resource utilization graph.

(b) Write a script which will shows all running process when your Linux system boots up.

Programs:

a)

```
import psutil
import time
import matplotlib.pyplot as plt

# create empty lists to store data for plotting
cpu_data = []
mem_data = []
disk_data = []
sent_data = []
recv_data = []

# set the plot title and labels
plt.title('System Resource Monitor')
plt.xlabel('Time (s)')
plt.ylabel('Usage (%)')

# set the plot axis limits
plt.ylim(0, 100)

# start the loop to monitor system resources
while True:
    # get the system resource usage
    cpu_percent = psutil.cpu_percent()
    mem_percent = psutil.virtual_memory().percent
    disk_percent = psutil.disk_usage('/').percent
    net_io_counters = psutil.net_io_counters()
    sent_mb = net_io_counters.bytes_sent / 1024 / 1024
    recv_mb = net_io_counters.bytes_recv / 1024 / 1024

    # add the data to the lists for plotting
    cpu_data.append(cpu_percent)
    mem_data.append(mem_percent)
    disk_data.append(disk_percent)
    sent_data.append(sent_mb)
    recv_data.append(recv_mb)

    # print the system resource usage
    print(f"CPU Usage: {cpu_percent}%")
    print(f"Memory Usage: {mem_percent}%")
    print(f"Disk Usage: {disk_percent}%")
    print(f"Network Usage: Sent: {sent_mb} MB, Received: {recv_mb} MB")

    plt.clf()
    plt.subplot(2, 2, 1)
    plt.plot(cpu_data, 'r-', label='CPU Usage')
    plt.ylabel('Usage (%)')
    plt.legend(loc='upper left')

    plt.subplot(2, 2, 2)
    plt.plot(mem_data, 'b-', label='Memory Usage')
    plt.ylabel('Usage (%)')
    plt.legend(loc='upper left')

    plt.subplot(2, 2, 3)
    plt.plot(disk_data, 'g-', label='Disk Usage')
    plt.ylabel('Usage (%)')
    plt.legend(loc='upper left')

    plt.subplot(2, 2, 4)
    plt.plot(sent_data, 'm-', label='Sent MB')
    plt.plot(recv_data, 'y-', label='Received MB')
    plt.ylabel('Usage (MB)')
    plt.legend(loc='upper left')

    plt.suptitle('System Resource Monitor')
    plt.pause(1)
    plt.show(block=False)
    time.sleep(1)
```

b)

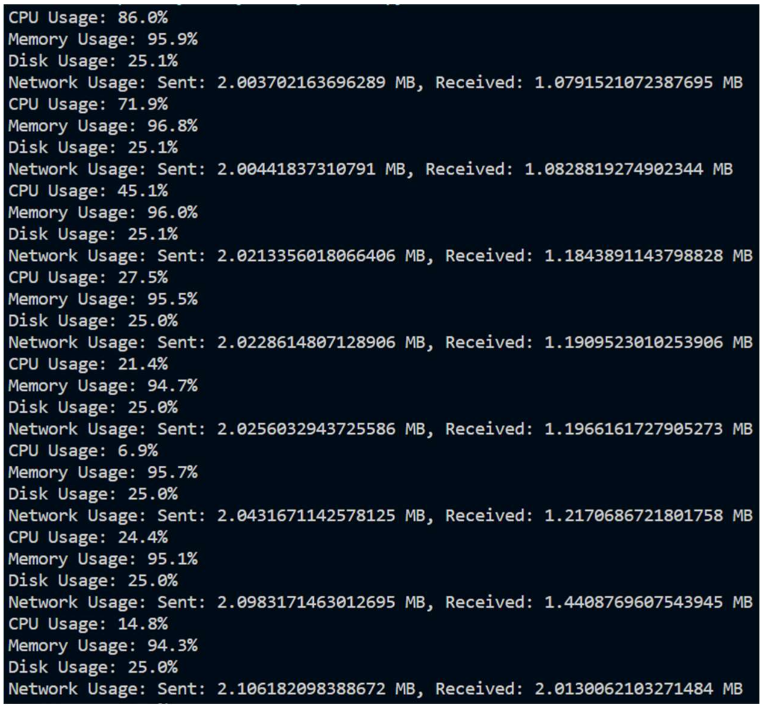
```
#!/bin/bash

echo "Listing all running processes at system bootup..."

while true; do
    ps -eo pid,ppid,cmd,%mem,%cpu --sort=-%mem | head
    sleep 5
done
```

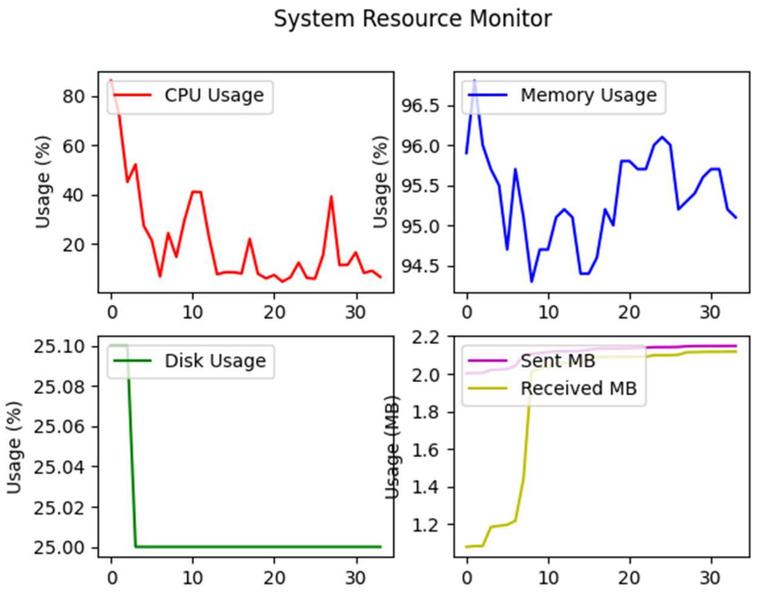
<u>Outputs:</u>

a) CODE OUTPUT:

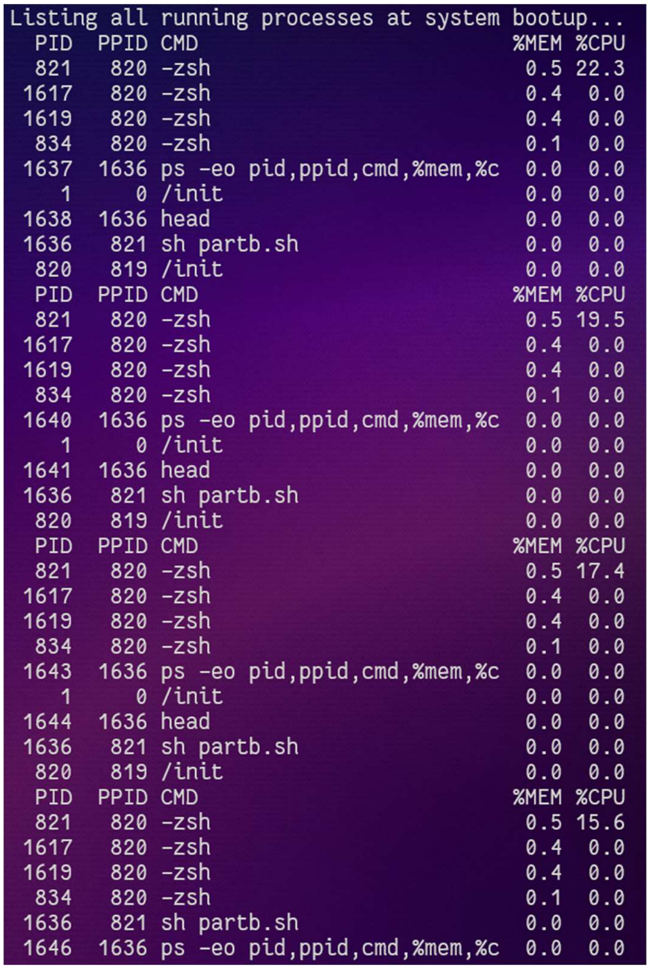


```
CPU Usage: 86.0%
Memory Usage: 95.9%
Disk Usage: 25.1%
Network Usage: Sent: 2.003702163696289 MB, Received: 1.0791521072387695 MB
CPU Usage: 71.9%
Memory Usage: 96.8%
Disk Usage: 25.1%
Network Usage: Sent: 2.00441837310791 MB, Received: 1.0828819274902344 MB
CPU Usage: 45.1%
Memory Usage: 96.0%
Disk Usage: 25.1%
Network Usage: Sent: 2.0213356018066406 MB, Received: 1.1843891143798828 MB
CPU Usage: 27.5%
Memory Usage: 95.5%
Disk Usage: 25.0%
Network Usage: Sent: 2.0228614807128906 MB, Received: 1.1909523010253906 MB
CPU Usage: 21.4%
Memory Usage: 94.7%
Disk Usage: 25.0%
Network Usage: Sent: 2.0256032943725586 MB, Received: 1.1966161727905273 MB
CPU Usage: 6.9%
Memory Usage: 95.7%
Disk Usage: 25.0%
Network Usage: Sent: 2.0431671142578125 MB, Received: 1.2170686721801758 MB
CPU Usage: 24.4%
Memory Usage: 95.1%
Disk Usage: 25.0%
Network Usage: Sent: 2.0983171463012695 MB, Received: 1.4408769607543945 MB
CPU Usage: 14.8%
Memory Usage: 94.3%
Disk Usage: 25.0%
Network Usage: Sent: 2.106182098388672 MB, Received: 2.0130062103271484 MB
```

GRAPH:



b) CODE OUTPUT:



```
Listing all running processes at system bootup...
  PID  PPID CMD                          %MEM %CPU
  821   820 -zsh                          0.5 22.3
 1617   820 -zsh                          0.4  0.0
 1619   820 -zsh                          0.4  0.0
  834   820 -zsh                          0.1  0.0
 1637  1636 ps -eo pid,ppid,cmd,%mem,%c  0.0  0.0
    1     0 /init                         0.0  0.0
 1638  1636 head                          0.0  0.0
 1636   821 sh partb.sh                   0.0  0.0
  820   819 /init                         0.0  0.0
  PID  PPID CMD                          %MEM %CPU
  821   820 -zsh                          0.5 19.5
 1617   820 -zsh                          0.4  0.0
 1619   820 -zsh                          0.4  0.0
  834   820 -zsh                          0.1  0.0
 1640  1636 ps -eo pid,ppid,cmd,%mem,%c  0.0  0.0
    1     0 /init                         0.0  0.0
 1641  1636 head                          0.0  0.0
 1636   821 sh partb.sh                   0.0  0.0
  820   819 /init                         0.0  0.0
  PID  PPID CMD                          %MEM %CPU
  821   820 -zsh                          0.5 17.4
 1617   820 -zsh                          0.4  0.0
 1619   820 -zsh                          0.4  0.0
  834   820 -zsh                          0.1  0.0
 1643  1636 ps -eo pid,ppid,cmd,%mem,%c  0.0  0.0
    1     0 /init                         0.0  0.0
 1644  1636 head                          0.0  0.0
 1636   821 sh partb.sh                   0.0  0.0
  820   819 /init                         0.0  0.0
  PID  PPID CMD                          %MEM %CPU
  821   820 -zsh                          0.5 15.6
 1617   820 -zsh                          0.4  0.0
 1619   820 -zsh                          0.4  0.0
  834   820 -zsh                          0.1  0.0
 1636   821 sh partb.sh                   0.0  0.0
 1646  1636 ps -eo pid,ppid,cmd,%mem,%c  0.0  0.0
```

Observation: The outputs are shown in the above snapshots

Conclusion: The programs are successfully executed.