Exp 13.

Date.

AIM :-

To implement your own ping program

ALGORITHM :-

UDP Server

→ Create UDP Socket & bind it to a specific address & port

→ wait for msg

→ Print msg & client address

→ Send back ping to client.

UDP client

→ Create UDP socket & set a recv timeout.

→ Send "ping" to server

→ If a response received - print response & calculate RTT.

→ If no response within recv timeout print request timeout.

CODE :-

Server.py

```
import socket
def start_server(host='127.0.0.1', port=12345):
    with socket.socket(socket.AF_INET, socket.SOCK_DGRAM) as s:
        s.bind((host, port))
        print("UDP Server running on {host}:{port}")
        while True:
            data, addr = s.recvfrom(1024)
            print(f"Received Message from {addr}: {data.decode()}")
            s.sendto(b'pong', addr)
if __name__ == "__main__":
    start_server()
```

client.py.

```
import time
import socket
def ping_server(host='127.0.0.1', port=12345):
    with socket.socket(socket.AF_INET, socket.SOCK_DGRAM
        try:
            s.settimeout(2)
            start = time.time()
            s.sendto(b'ping', (host, port))
            end=time.time()
            print(f"Received {data.decode()} from {addr}
in {end-start:.2f} seconds")
        except socket.timeout:
            print("Request timed out")
if __name__ == "__main__":
    ping_server()
```

OUTPUT

Terminal
python server.py
UDP server running
on 127.0.0.1:12345

Received Msg.
'127.0.0.1' 150061 : ping

Terminal.

python client.py
Received pong from ('127.0.0.1',
12345) in 0.00 sec)