Roll No:241901097 Name: SANGATHAMIZHAN.S.P

EXPT: 4 DEVELOP A CUSTOMIZED PING COMMAND TO TEST THE SERVER CONNECTIVITY

AIM:

Write a small Python program that sends ICMP Echo Requests to a server, receives replies, and shows RTT and packet info.

ALGORITHM:

1. Start
2. Import required modules:
    - socket for network communication
    - time for measuring delay
3. Set the target host (e.g., "google.com").
4. Set the port number to 80 (HTTP port).
5. Set the number of attempts (count = 4).
6. Create an empty list times to store RTT values.
7. Repeat the following steps **count** times:
    - Use a **try block** to handle errors.
    - Create a socket object.
    - Record start time using time.time().
    - Attempt to connect to the given host and port using connect().
    - Record end time using time.time().
    - Close the socket.
    - Calculate RTT using: **RTT = (end – start) × 1000** (convert to milliseconds).Store RTT in the list. Display the reply time for the attempt.
8. If any RTT values are stored:
    - Compute **minimum RTT** using min(times).
    - Compute **maximum RTT** using max(times).
    - Compute **average RTT** using sum(times)/len(times).
    - Display all computed values.
9. Stop

CODE:

```
import socket, time
```

```
host = "google.com"
port = 80
count = 4
times = []
for i in range(count):
    try:
        s = socket.socket()
        start = time.time()
        s.connect((host, port))
        end = time.time()
        s.close()
        rtt = (end - start) * 1000
        times.append(rtt)
        print(f"Reply from {host}: time={rtt:.2f} ms")
    except:
        print("Request timed out")
if times:
    print("\nMin RTT =", min(times), "ms")
    print("Max RTT =", max(times), "ms")
    print("Avg RTT =", sum(times)/len(times), "ms")
```

OUTPUT:

RESULT: The custom ping program successfully sent ICMP Echo Requests and received Echo Replies from the target.