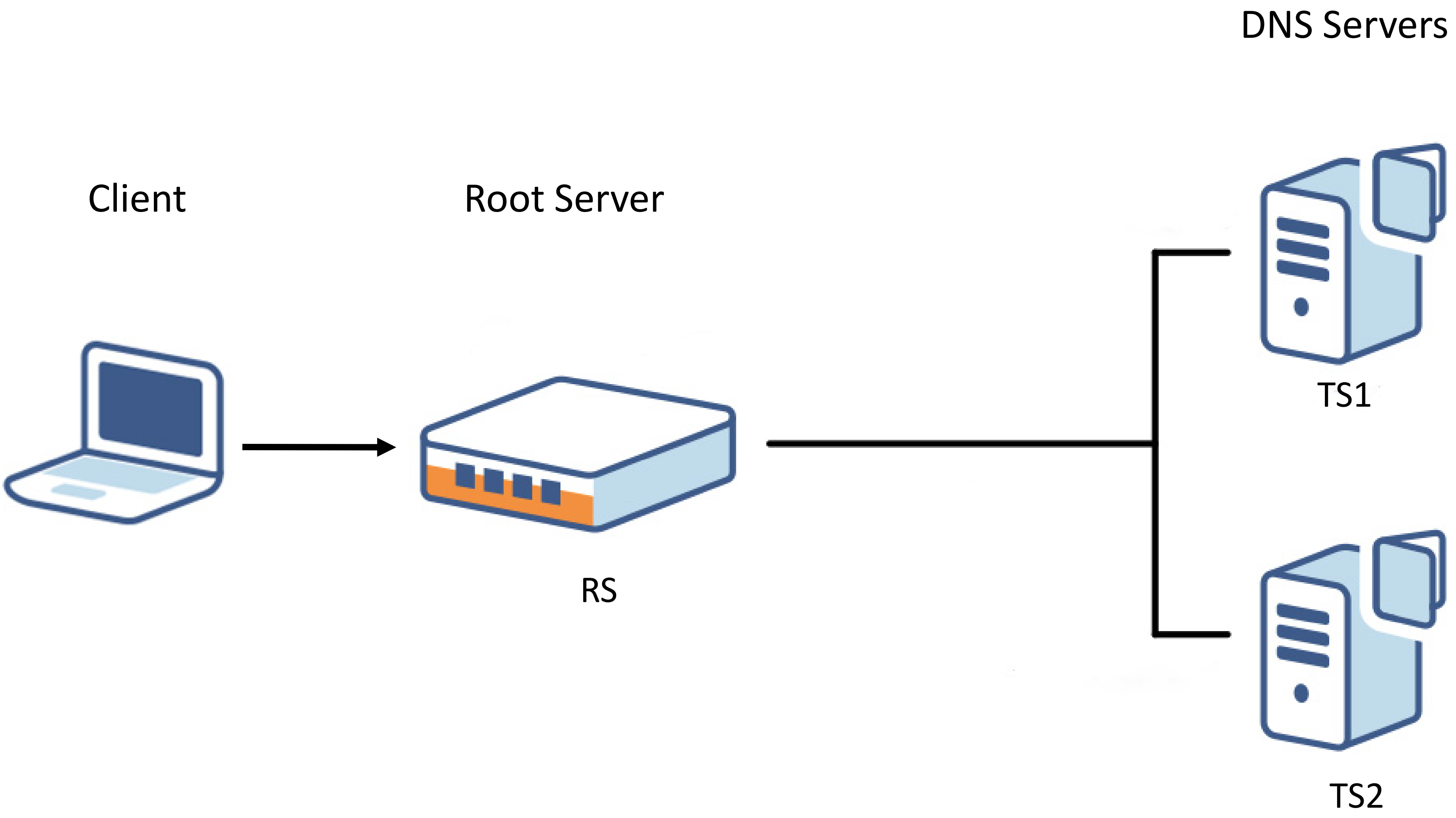
Internet Technology
Recitation Section 03

Negin Dehghanchaleshtori

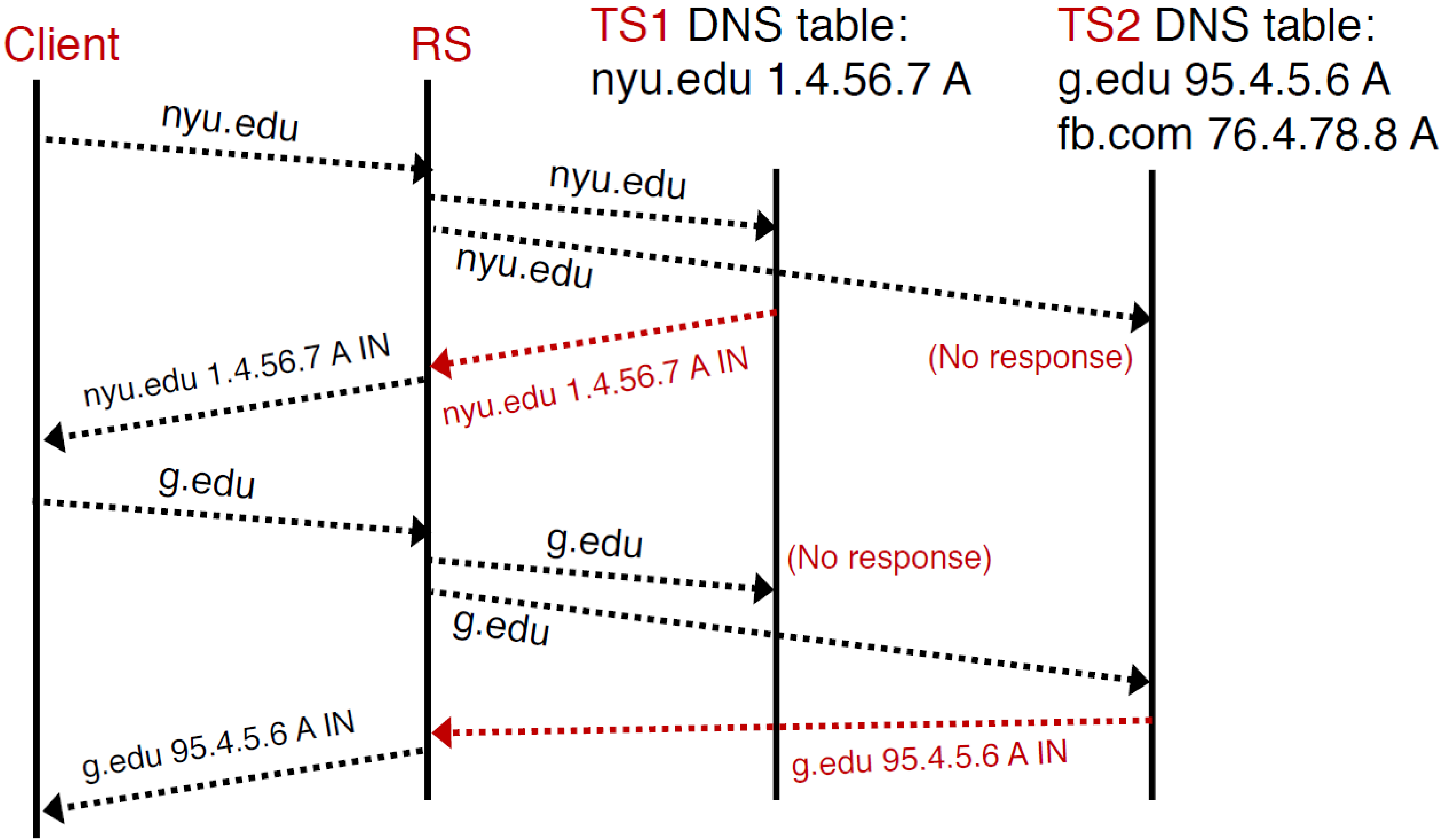
Computer Science Department

Spring 2022

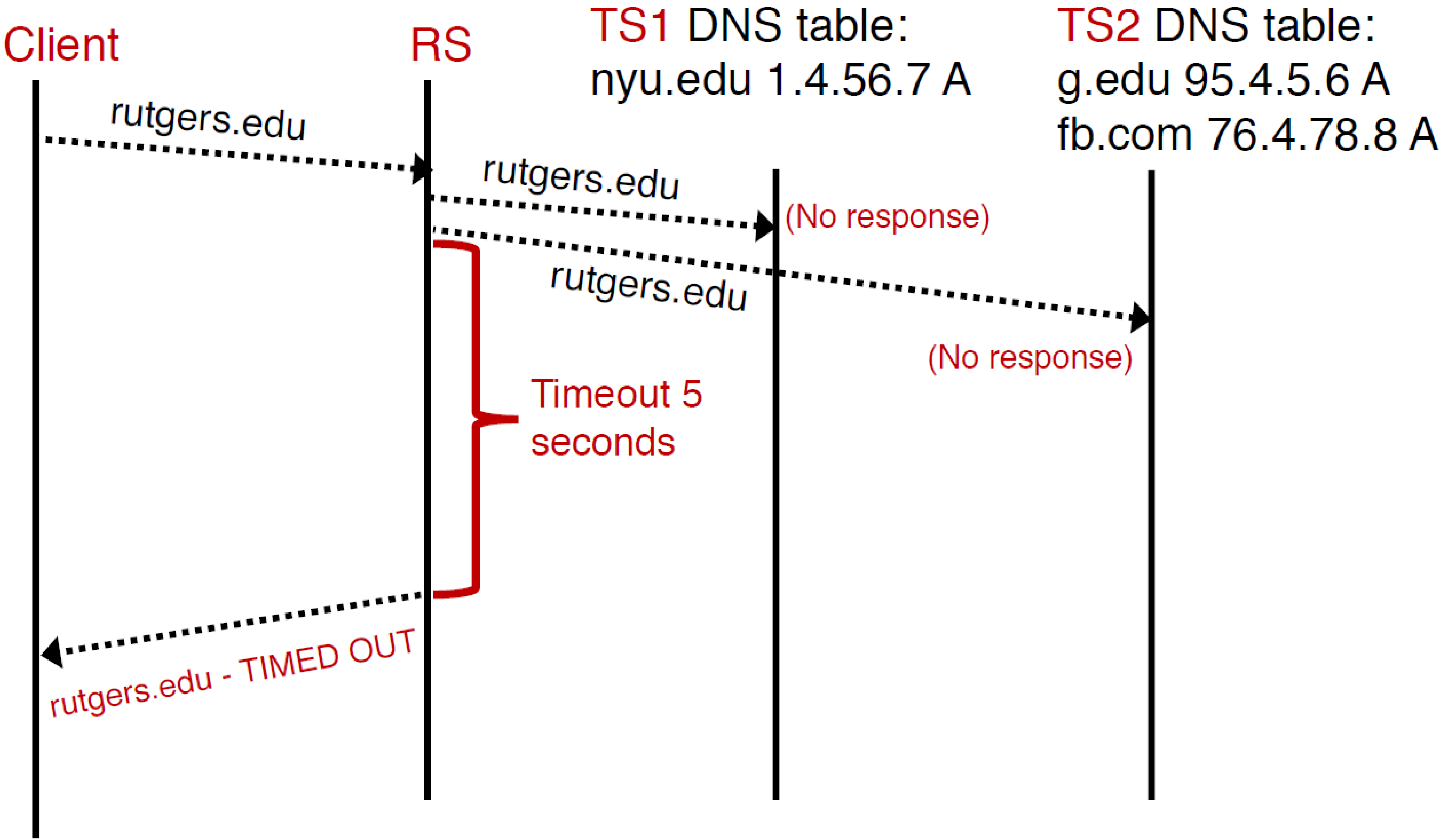
Load Balancing across DNS Server



Load Balancing across DNS Server



Load Balancing across DNS Server



Design of TS Server

Look up a domain name queried by RS and return entry with an IP address

DNS Table

- Domain Name
- IP address
- Type (A only in this project)

two machines establish a connection, both must create a socket object

Design of TS Server

Response

DomainName IPaddress A IN

- Name
- Value
- Type
- Class

DNS Table

PROJ2-DNSTS1.txt
PROJ2-DNSTS2.txt

```
www.princeton.edu 128.1.1.4 A
GOOGLE.com 46.1.3.7 A
```

Design of RS Server

Receive and forward query to TS1 & TS2
Relay response to client

Hostname IPaddress A IN

DomainName - TIMED OUT

RS Server connections

One with each TS

One with the Client

Client

Read domain names form PROJ2-HNS.txt

Write outputs it receives into RESOLVED.txt

Only one connection with RS

What you must submit

rs.py
ts1.py
ts2.py
client.py
report.pdf

What you must submit

Names and netids

Resources and references

RS functionality that tracks which TS responded to a given query or timing out

Does not work

Difficulties

What did you learn
any interesting observations

Notes

Teams of two

Discuss on Piazza

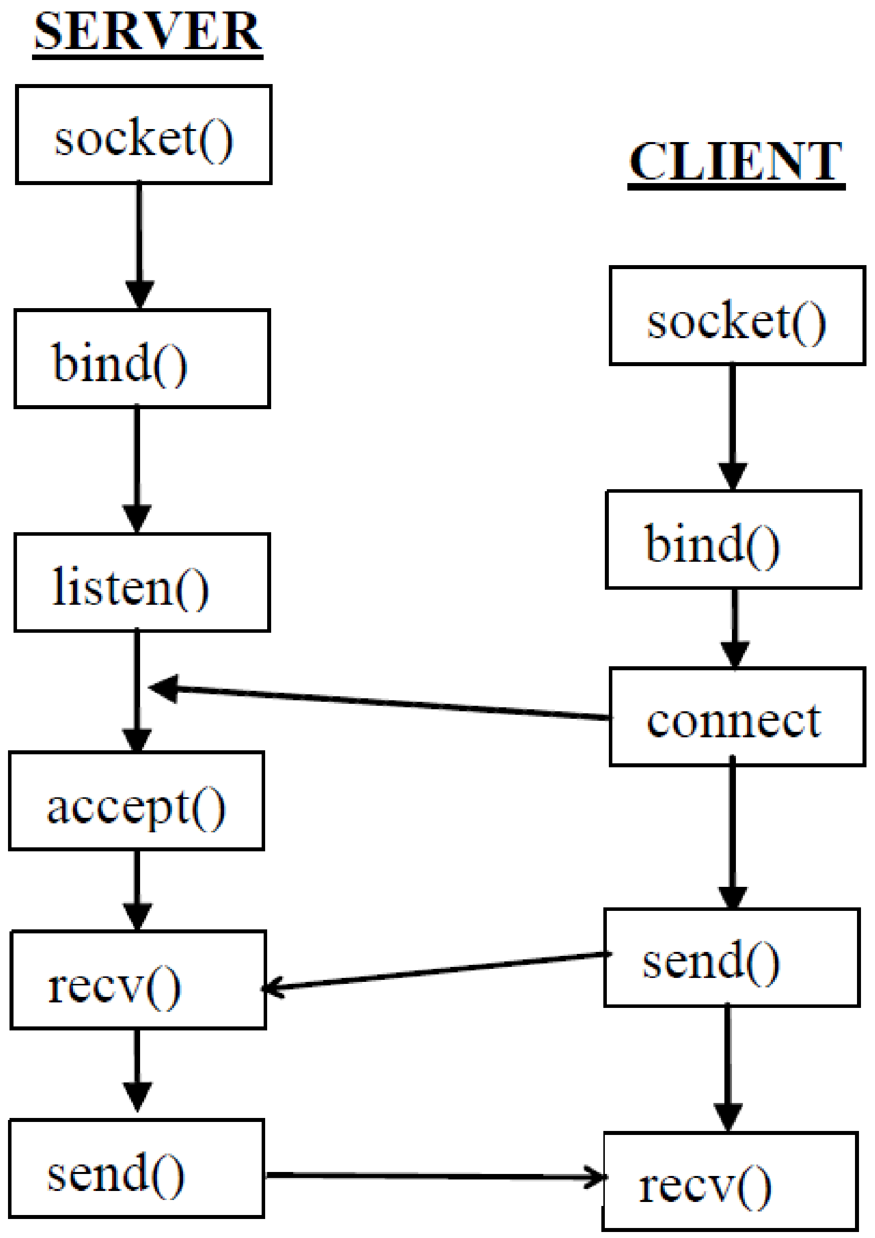
Do not copy, Do not post project

Be clear

Include people and resources

Due 17 Feb

TCP Flow Chart



Create Server

- **s.bind()**

  This method binds address (hostname, port number pair) to socket.

- **s.listen()**

  This method sets up and start listener.

- **s.accept()**

  This passively accept client connection, waiting until connection arrives

s.bind((socket.gethostname(), 1234))

Tuple of hostname and port number

Create Client

- **s.connect()**

  This method actively initiates server connection.

General Methods

- **s.recv()**
  This method receives TCP message
- **s.send()**
  This method transmits TCP message
- **s.recvfrom()**
  This method receives UDP message
- **s.sendto()**
  This method transmits UDP message
- **s.close()**
  This method closes socket
- **socket.gethostname()**
  Returns the hostname.

RS Server connections

One with each TS

One with the Client

Most tricky part

figuring out which TS

~~*re[illegible]()*~~

Blocking Function

Has to wait for something to complete

setblocking(0)
Never wait for the operation to complete
Just put as much data as possible

Select Module

Deal with multiple file descriptors at once

select
poll
epoll
kqueue

Select Module

Three arguments

List of file descriptors to watch for reading
List of file descriptors to watch for writing
List of file descriptors to watch for errors

Timeout optional 4th argument