



LOYOLA – ICAM

COLLEGE OF ENGINEERING & TECHNOLOGY (LICET)

Loyola Campus, Nungambakkam, Chennai – 600 034
Tamil Nadu, India

Approved by AICTE and
Affiliated to Anna University, Chennai
A Christian Minority Institution
ISO 9001:2008 Certified Institution

RECORD NOTE BOOK

Name : _____

Register No : _____

Year / Semester : _____

Branch : _____

Subject Code & Name : _____



LOYOLA – ICAM

COLLEGE OF ENGINEERING & TECHNOLOGY (LICET)

Loyola Campus, Nungambakkam, Chennai – 600 034
Tamil Nadu, India

Certificate

Name :

Register No. :

*Certified that this is a bonafide record of work done by the candidate in the
..... semester of B.E/B.Tech.....
in the Laboratory during the year*

Lab. In-Charge

Head of the Department

*This record is submitted for Anna University practical examination held on
.....*

Internal Examiner

External Examiner

INDEX

Expt. No.	Date	Title of the Experiment	Submission Date	Marks



LOYOLA – ICAM
COLLEGE OF ENGINEERING AND TECHNOLOGY (LICET)
Loyola College Campus, Nungambakkam, Chennai – 34

DEPARTMENT OF INFORMATION TECHNOLOGY

Vision of the Department:

To build proficient Information Technologists through moral, ethical and technological standards for the societal well-being.

Mission of the Department:

- To provide a practice oriented methodology with access to contemporary knowledge in Information Technology for the betterment of the society
- To prepare students for competent careers in Information Technology through innovation, team spirit, ethics and entrepreneurial skills in evolving technologies
- To integrate our department with quality organizations worldwide and promote industry institute interaction for symbiotic benefits
- To promote interdisciplinary research through innovation and reflective thinking

Program Educational Objectives:

PEO1: Graduates will possess the ability to apply their technological skills to comprehend and analyze complex problems to design and implement the feasible solutions.

PEO2: Graduates will acquire the desire for lifelong learning and ability to work in multidisciplinary teams for meeting the global challenges

PEO3: Graduates will be able to exhibit professional ethics, skills for management and responsibility towards societal needs.

Program Specific Outcome:

Engineering Graduates will able to:

- PSO1** Apply technological advances in the field of Information Technology for societal issues through professional ethics.
- PSO2** Acquire design skills for conducting domain specific experiments and interpreting data to synthesize and analyze information
- PSO3** Deploy appropriate algorithms, latest open source softwares and other related programming engineering applications

Aim:

To use commands like tracert, ping, ipconfig, Nslookup and netstat PDU'S using a network protocol analyzer.

Basic commands:

arp -a: ARP is short form of address relation protocol, It will show the IP address of your computer along with the IP address and mac address of your router.

```
C:\Users\Nithila>arp -a

Interface: 192.168.29.247 --- 0x13
    Internet Address      Physical Address      Type
    192.168.29.1          a8-da-0c-4b-b4-ae     dynamic
    224.0.0.22            01-00-5e-00-00-16     static
    224.0.0.251           01-00-5e-00-00-fb     static
    224.0.0.252           01-00-5e-00-00-fc     static
    239.255.255.250       01-00-5e-7f-ff-fa     static
    255.255.255.255       ff-ff-ff-ff-ff-ff     static
```

hostname: This is the simplest of all TCP/IP commands. It simply displays the name of your computer.

```
C:\Users\Nithila>hostname
DESKTOP-KLA6B2R
```

ipconfig: The ipconfig command displays information about the host (the computer we're operating with) computer TCP/IP configuration.

```

C:\Users\Nithila>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 1:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 2:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Wi-Fi:

    Connection-specific DNS Suffix  . :
    IPv6 Address. . . . . : 2405:201:e025:1033:7d35:4895:26de:efbd
    Temporary IPv6 Address. . . . . : 2405:201:e025:1033:4a8:83f:2e08:8f78
    Link-local IPv6 Address . . . . . : fe80::7d35:4895:26de:efbd%19
    IPv4 Address. . . . . : 192.168.29.247
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : fe80::aada:cff:fe4b:b4ae%19
                                192.168.29.1

Ethernet adapter Bluetooth Network Connection:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

```

ipconfig /all: This command displays detailed configuration information about TCP/IP configuration.

```

C:\Users\Nithila>ipconfig /all

Windows IP Configuration

    Host Name . . . . . : DESKTOP-KLA6B2R
    Primary Dns Suffix . . . . . :
    Node Type . . . . . : Hybrid
    IP Routing Enabled. . . . . : No
    WINS Proxy Enabled. . . . . : No

Ethernet adapter Ethernet:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :
    Description . . . . . : Realtek PCIe GbE Family Controller
    Physical Address. . . . . : 70-B5-E8-C1-C7-6C
    DHCP Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . : Yes

Wireless LAN adapter Local Area Connection* 1:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :
    Description . . . . . : Microsoft Wi-Fi Direct Virtual Adapter
    Physical Address. . . . . : 34-CF-F6-8E-2D-09
    DHCP Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . : Yes

Wireless LAN adapter Local Area Connection* 2:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :
    Description . . . . . : Microsoft Wi-Fi Direct Virtual Adapter #2
    Physical Address. . . . . : 36-CF-F6-8E-2D-08
    DHCP Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . : Yes

```

```

Wireless LAN adapter Wi-Fi:

    Connection-specific DNS Suffix . : 
    Description . . . . . : Intel(R) Wireless-AC 9462
    Physical Address. . . . . : 34-CF-F6-8E-2D-08
    DHCP Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . : Yes
    IPv6 Address. . . . . : 2405:201:e025:1033:7d35:4895:26de:efbd(Preferred)
    Temporary IPv6 Address. . . . . : 2405:201:e025:1033:4a8:83f:2e08:8f78(Preferred)
    Link-local IPv6 Address . . . . . : fe80::7d35:4895:26de:efbd%19(Preferred)
    IPv4 Address. . . . . : 192.168.29.247(Preferred)
    Subnet Mask . . . . . : 255.255.255.0
    Lease Obtained. . . . . : 22 August 2021 20:14:47
    Lease Expires . . . . . : 23 August 2021 00:34:32
    Default Gateway . . . . . : fe80::aada:cff:fe4b:b4ae%19
                                192.168.29.1
    DHCP Server . . . . . : 192.168.29.1
    DHCPv6 IAID . . . . . : 171233270
    DHCPv6 Client DUID. . . . . : 00-01-00-01-26-AB-F9-0B-70-B5-E8-C1-C7-6C
    DNS Servers . . . . . : 2405:201:e025:1033::c0a8:1d01
                                192.168.29.1
    NetBIOS over Tcpip. . . . . : Enabled

Ethernet adapter Bluetooth Network Connection:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix . : 
    Description . . . . . : Bluetooth Device (Personal Area Network)
    Physical Address. . . . . : 34-CF-F6-8E-2D-0C
    DHCP Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . : Yes

```

ipconfig /renew: Using this command will renew all your IP address that you're currently borrowing from DHCP server. This command is a quick problem solver if you are having connection issue, but does not work if you have been configured with static IP address

```
C:\Users\Nithila>ipconfig /renew
```

Windows IP Configuration

```
No operation can be performed on Ethernet while it has its media disconnected.  
No operation can be performed on Local Area Connection* 1 while it has its media disconnected.  
No operation can be performed on Local Area Connection* 2 while it has its media disconnected.  
No operation can be performed on Bluetooth Network Connection while it has its media disconnected.
```

Ethernet adapter Ethernet:

```
Media State . . . . . : Media disconnected  
Connection-specific DNS Suffix . :
```

Wireless LAN adapter Local Area Connection* 1:

```
Media State . . . . . : Media disconnected  
Connection-specific DNS Suffix . :
```

Wireless LAN adapter Local Area Connection* 2:

```
Media State . . . . . : Media disconnected  
Connection-specific DNS Suffix . :
```

Wireless LAN adapter Wi-Fi:

```
Connection-specific DNS Suffix . :  
IPv6 Address. . . . . : 2405:201:e025:1033:7d35:4895:26de:efbd  
Temporary IPv6 Address. . . . . : 2405:201:e025:1033:4a8:83f:2e08:8f78  
Link-local IPv6 Address . . . . . : fe80::7d35:4895:26de:efbd%19  
IPv4 Address. . . . . : 192.168.29.247  
Subnet Mask . . . . . : 255.255.255.0  
Default Gateway . . . . . : fe80::aada:cff:fe4b:b4ae%19  
                            192.168.29.1
```

Ethernet adapter Bluetooth Network Connection:

```
Media State . . . . . : Media disconnected  
Connection-specific DNS Suffix . :
```


ipconfig /release: ipconfig /release <adapter> - Releases the IPv4 Address for a Specific Network Adapters. The /release <adapter> switch will cause ipconfig to drop the dynamically issued IPv4 address by sending a DHCPRELEASE message to the DHCP server for a specific network adapter.

```
C:\Users\Nithila>ipconfig /release

Windows IP Configuration

No operation can be performed on Ethernet while it has its media disconnected.
No operation can be performed on Local Area Connection* 1 while it has its media disconnected.
No operation can be performed on Local Area Connection* 2 while it has its media disconnected.
No operation can be performed on Bluetooth Network Connection while it has its media disconnected.

Ethernet adapter Ethernet:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 1:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 2:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Wi-Fi:

    Connection-specific DNS Suffix  . :
    IPv6 Address. . . . . : 2405:201:e025:1033:7d35:4895:26de:efbd
    Temporary IPv6 Address. . . . . : 2405:201:e025:1033:4a8:83f:2e08:8f78
    Link-local IPv6 Address . . . . . : fe80::7d35:4895:26de:efbd%19
    Default Gateway . . . . . : fe80::aada:cff:fe4b:b4ae%19

Ethernet adapter Bluetooth Network Connection:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

C:\Users\Nithila>
```

netstat: netstat displays network connection for TCP (both incoming and outgoing), routing tables, a number of network interface and a network protocol.

```
C:\Users\Nithila>netstat

Active Connections

Proto Local Address           Foreign Address         State
TCP    127.0.0.1:49672          DESKTOP-KLA6B2R:49673  ESTABLISHED
TCP    127.0.0.1:49673          DESKTOP-KLA6B2R:49672  ESTABLISHED
TCP    127.0.0.1:49674          DESKTOP-KLA6B2R:49675  ESTABLISHED
TCP    127.0.0.1:49675          DESKTOP-KLA6B2R:49674  ESTABLISHED
TCP    192.168.29.247:49392     1drv:https              ESTABLISHED
TCP    192.168.29.247:49475     20.197.71.89:https      ESTABLISHED
TCP    192.168.29.247:50982     reliance:domain          SYN_SENT
TCP    192.168.29.247:53621     219:https                TIME_WAIT
TCP    192.168.29.247:53634     ec2-54-149-217-243:https TIME_WAIT
TCP    192.168.29.247:55519     219:https                TIME_WAIT
TCP    192.168.29.247:55520     20.197.71.89:https      ESTABLISHED
TCP    192.168.29.247:55522     1drv:https              TIME_WAIT
TCP    192.168.29.247:64713     23.98.104.196:https     ESTABLISHED
TCP    192.168.29.247:64714     162.159.135.234:https   ESTABLISHED
TCP    [2405:201:e025:1033:4a8:83f:2e08:8f78]:53619 whatsapp-cdn6-shv-01-tir2:https ESTABLISHED
TCP    [2405:201:e025:1033:4a8:83f:2e08:8f78]:56307 maa05s18-in-x05:https   TIME_WAIT
TCP    [2405:201:e025:1033:4a8:83f:2e08:8f78]:56314 g2600-140f-0400-0195-0000-0000-0000-4106:https CLOSE_WAIT
TCP    [2405:201:e025:1033:4a8:83f:2e08:8f78]:56317 g2600-140f-0400-01ac-0000-0000-0000-3114:http CLOSE_WAIT
TCP    [2405:201:e025:1033:4a8:83f:2e08:8f78]:56318 g2600-140f-0400-01ac-0000-0000-0000-3114:http CLOSE_WAIT
TCP    [2405:201:e025:1033:4a8:83f:2e08:8f78]:56319 g2600-140f-0400-01ac-0000-0000-0000-3114:http CLOSE_WAIT
TCP    [2405:201:e025:1033:4a8:83f:2e08:8f78]:56320 g2600-140f-0400-01ac-0000-0000-0000-3114:http CLOSE_WAIT
TCP    [2405:201:e025:1033:4a8:83f:2e08:8f78]:56321 g2600-140f-0400-01ac-0000-0000-0000-3114:http CLOSE_WAIT
TCP    [2405:201:e025:1033:4a8:83f:2e08:8f78]:56339 [2606:2800:147:120f:30c:1ba0:fc6:265a]:https CLOSE_WAIT
TCP    [2405:201:e025:1033:4a8:83f:2e08:8f78]:64712 [2404:6800:4003:c03::bc]:5228 ESTABLISHED
```

Nslookup: queries the specifies DNS server and retrievers the requested the request associated with the domain name you provided. These records contain interface domain names IP address.

```
C:\Users\Nithila>Nslookup
Default Server:  reliance.reliance
Address:  2405:201:e025:1033::c0a8:1d01

> exit
```

pathping: The pathping command is a command-line network utility that combines the functionality of ping with that of tracert. It is used to locate spots that have network latency and network loss.

```
C:\Users\Nithila>pathping

Usage: pathping [-g host-list] [-h maximum_hops] [-i address] [-n]
               [-p period] [-q num_queries] [-w timeout]
               [-4] [-6] target_name

Options:
  -g host-list      Loose source route along host-list.
  -h maximum_hops   Maximum number of hops to search for target.
  -i address        Use the specified source address.
  -n               Do not resolve addresses to hostnames.
  -p period         Wait period milliseconds between pings.
  -q num_queries    Number of queries per hop.
  -w timeout        Wait timeout milliseconds for each reply.
  -4               Force using IPv4.
  -6               Force using IPv6.
```

ping: The ping command is a Command Prompt command used to test the ability of the source computer to reach a specified destination computer. It's usually used as a simple way to verify that a computer can communicate over the network with another computer or network device.

```
C:\Users\Nithila>ping

Usage: ping [-t] [-a] [-n count] [-l size] [-f] [-i TTL] [-v TOS]
           [-r count] [-s count] [[-j host-list] | [-k host-list]]
           [-w timeout] [-R] [-S srcaddr] [-c compartment] [-p]
           [-4] [-6] target_name

Options:
    -t                Ping the specified host until stopped.
                     To see statistics and continue - type Control-Break;
                     To stop - type Control-C.
    -a                Resolve addresses to hostnames.
    -n count          Number of echo requests to send.
    -l size           Send buffer size.
    -f                Set Don't Fragment flag in packet (IPv4-only).
    -i TTL            Time To Live.
    -v TOS            Type Of Service (IPv4-only. This setting has been deprecated
                     and has no effect on the type of service field in the IP
                     Header).
    -r count          Record route for count hops (IPv4-only).
    -s count          Timestamp for count hops (IPv4-only).
    -j host-list       Loose source route along host-list (IPv4-only).
    -k host-list       Strict source route along host-list (IPv4-only).
    -w timeout        Timeout in milliseconds to wait for each reply.
    -R                Use routing header to test reverse route also (IPv6-only).
                     Per RFC 5095 the use of this routing header has been
                     deprecated. Some systems may drop echo requests if
                     this header is used.
    -S srcaddr        Source address to use.
    -c compartment    Routing compartment identifier.
    -p                Ping a Hyper-V Network Virtualization provider address.
    -4                Force using IPv4.
    -6                Force using IPv6.
```

tracert: The tracert command (spelled traceroute in Unix/Linux implementations) is one of the key diagnostic tools for TCP/IP. It displays a list of all the routers that a packet must go through to get from the computer where tracert is run to any other computer on the Internet.

```
C:\Users\Nithila>tracert

Usage: tracert [-d] [-h maximum_hops] [-j host-list] [-w timeout]
              [-R] [-S srcaddr] [-4] [-6] target_name

Options:
  -d                Do not resolve addresses to hostnames.
  -h maximum_hops   Maximum number of hops to search for target.
  -j host-list       Loose source route along host-list (IPv4-only).
  -w timeout         Wait timeout milliseconds for each reply.
  -R                Trace round-trip path (IPv6-only).
  -S srcaddr         Source address to use (IPv6-only).
  -4                Force using IPv4.
  -6                Force using IPv6.
```

route: The route command allows you to make manual entries into the network routing tables. It distinguishes between routes to hosts and routes to networks by interpreting the network address of the Destination variable, which can be specified either by symbolic name or numeric address.


```
C:\Users\Nithila>route
```

Manipulates network routing tables.

```
ROUTE [-f] [-p] [-4|-6] command [destination]
      [MASK netmask] [gateway] [METRIC metric] [IF interface]
```

-f	Clears the routing tables of all gateway entries. If this is used in conjunction with one of the commands, the tables are cleared prior to running the command.
-p	When used with the ADD command, makes a route persistent across boots of the system. By default, routes are not preserved when the system is restarted. Ignored for all other commands, which always affect the appropriate persistent routes.
-4	Force using IPv4.
-6	Force using IPv6.
command	One of these: PRINT Prints a route ADD Adds a route DELETE Deletes a route CHANGE Modifies an existing route
destination	Specifies the host.
MASK	Specifies that the next parameter is the 'netmask' value.
netmask	Specifies a subnet mask value for this route entry. If not specified, it defaults to 255.255.255.255.
gateway	Specifies gateway.
interface	the interface number for the specified route.
METRIC	specifies the metric, ie. cost for the destination.

All symbolic names used for destination are looked up in the network database file NETWORKS. The symbolic names for gateway are looked up in the host name database file HOSTS.

If the command is PRINT or DELETE. Destination or gateway can be a wildcard, (wildcard is specified as a star '*'), or the gateway argument may be omitted.

If Dest contains a * or ?, it is treated as a shell pattern, and only matching destination routes are printed. The '*' matches any string, and '?' matches any one char. Examples: 157.*.1, 157.*, 127.*, *224*.

```

Pattern match is only allowed in PRINT command.
Diagnostic Notes:
  Invalid MASK generates an error, that is when (DEST & MASK) != DEST.
  Example> route ADD 157.0.0.0 MASK 155.0.0.0 157.55.80.1 IF 1
           The route addition failed: The specified mask parameter is invalid. (Destination & Mask) != Destination.

Examples:

> route PRINT
> route PRINT -4
> route PRINT -6
> route PRINT 157*      .... Only prints those matching 157*

> route ADD 157.0.0.0 MASK 255.0.0.0 157.55.80.1 METRIC 3 IF 2
           destination^      ^mask      ^gateway      metric^      ^
                                   Interface^

  If IF is not given, it tries to find the best interface for a given
  gateway.
> route ADD 3ffe::/32 3ffe::1

> route CHANGE 157.0.0.0 MASK 255.0.0.0 157.55.80.5 METRIC 2 IF 2

  CHANGE is used to modify gateway and/or metric only.

> route DELETE 157.0.0.0
> route DELETE 3ffe::/32

```

Result:

The basic commands like tracert, ping, ipconfig, Nslookup and netstat PDU'S using a network protocol analyzer are executed and verified.

Aim:

To download a webpage using Java

Algorithm:**Client Side:**

- 1) Start the program.
- 2) Create a socket which binds the Ip address of the server and the port address to acquire service.
- 3) After establishing a connection, send the URL to the server.
- 4) Open a file and store the received data into the file.
- 5) Close the socket.
- 6) End the program.

Server Side:

- 1) Start the program.
- 2) Create a server socket to activate the port address.
- 3) Create a socket for the server socket which accepts the connection.
- 4) After establishing a connection, receive the URL from the client.
- 5) Download the content of the URL received and send the data to the client.
- 6) Close the socket.
- 7) End the program.

Source Code (Main123.java)

```
import java.io.*;
import java.net.*;
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.FileWriter;
import java.io.InputStreamReader;
import java.net.URL;

public class Main123 {
    public static void main(String[] args) throws Exception {
        URL url = new URL("https://ideathon.pattarai.in/");
        BufferedReader reader = new BufferedReader(new InputStreamReader(url.openStream()));
        BufferedWriter writer = new BufferedWriter(new FileWriter("data.html"));
        String line;
```

```

while ((line = reader.readLine()) != null) {
    System.out.println(line);
    writer.write(line);
    writer.newLine();
}
reader.close();
writer.close();
}
}

```

Output - Server Side:

```

C:\Program Files\Java\jdk-9.0.4\bin>Command Prompt
D:\Java>javac Main123.java

D:\Java>java Main123
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>IDEATHON II - By Licet Pattarai</title>

  <!-- METAS -->
  <meta property="og:title" content="Ideathon - By Licet Pattarai">
  <meta property="og:description"
    content="Pattarai is back with the platform for REINVENTING IMAGINATIONS of every imaginers out there. Find a problem, Come up with an impacting idea, Leave your mark!">
  <meta property="og:image"
    content="https://raw.githubusercontent.com/pattarai/pattarai-media/04a672dafce9e0acbb2be82747b6e92da5c6e99a/stock/SVG/gold_circle.svg">
  <meta property="og:url" content="https://ideathon.pattarai.in/">
  <meta name="twitter:title" content="Ideathon - By Licet Pattarai">
  <meta name="twitter:description"
    content="Pattarai is back with the platform for REINVENTING IMAGINATIONS of every imaginers out there. Find a problem, Come up with an impacting idea, Leave your mark!">
  <meta name="twitter:image"
    content="https://raw.githubusercontent.com/pattarai/pattarai-media/04a672dafce9e0acbb2be82747b6e92da5c6e99a/stock/SVG/gold_circle.svg">
  <meta name="twitter:site" content="@licetpattarai">
  <meta name="twitter:image:alt" content="Ideathon - By Licet Pattarai">
  <meta name="twitter:card"
    content="https://raw.githubusercontent.com/pattarai/pattarai-media/04a672dafce9e0acbb2be82747b6e92da5c6e99a/stock/SVG/gold_circle.svg">

  <link href="https://fonts.googleapis.com/css2?family=Source+Sans+Pro:wght@300;400&display=swap" rel="stylesheet">
  <link rel="stylesheet" href="css/bootstrap.min.css">
  <link rel="stylesheet" href="css/slick.css" type="text/css" />
  <link rel="stylesheet" href="css/templatemo-style.css">
  <link rel="stylesheet" href="css/modal.css">
  <link rel="icon" type="image/png"
    href="https://raw.githubusercontent.com/pattarai/pattarai-media/04a672dafce9e0acbb2be82747b6e92da5c6e99a/stock/SVG/gold_circle.svg">
</head>

<body>
  <video autoplay muted loop id="bg-video">
    <source src="video/gfp-astro-timelapse.mp4" type="video/mp4">
  </video>
  <div class="page-container">
    <div class="container-fluid">
      <div class="row">
        <div class="col-xs-12">
          <div class="cd-slider-nav">
            <nav class="navbar navbar-expand-lg id="tm-nav">

```



```

<a class="navbar-brand " style="font-weight: bold;" href="#">IDEATHON II</a>
<button class="navbar-toggler" type="button" data-bs-toggle="collapse"
  data-bs-target="#navbar-supported-content" aria-controls="navbarSupportedContent" aria-expanded="false"
  aria-label="Toggle navigation">
  <span class="navbar-toggler-icon"></span>
</button>
<div class="collapse navbar-collapse" id="navbar-supported-content">
  <ul class="navbar-nav mb-2 mb-lg-0">
    <li class="nav-item selected">
      <a class="nav-link text-center" aria-current="page" data-no="1" style="font-size:15px;">
        Home
      </a>
      <div class="circle"></div>
    </li>
    <li class="nav-item">
      <a class="nav-link text-center" data-no="2" style="font-size: 15px;">Guidelines</a>
      <div class="circle"></div>
    </li>
    <li class="nav-item">
      <a class="nav-link text-center" data-no="3" style="font-size: 15px;">Resources</a>
      <div class="circle"></div>
    </li>
    <li class="nav-item">
      <a class="nav-link text-center" data-no="4" style="font-size: 15px;">Register</a>
      <div class="circle"></div>
    </li>
  </ul>
</div>
</nav>
</div>
</div>
</div>
</div>
<div class="container-fluid tm-content-container">
  <ul class="cd-hero-slider mb-0 py-5">
    <li data-page-no="1">
      <div>
        <div class="d-flex position-relative tm-border-top tm-border-bottom intro-container">
          <div class="intro-left tm-bg-dark">
            
            <h1 class="mb-4 mt-4 pt-5" style="font-weight: bold; font-size: 3.0em;">Imagine. <span
              style="font-weight: bold; color: #FFC284;">Ideate. </span>Innovate.</h1>
            <p class="mb-4 ">
              <h4 class="text-uppercase" style="font-weight: lighter;"> Time to wear your
                Thinking Caps!</h4>
            <br>

            Pattarai is back with the platform for <span
              style="font-weight: bold; color: #FFC284;"></span>REINVENTING
            IMAGINATIONS</span>

```

```

of every imaginations out there.
Find a problem, Come up with an impacting idea, Leave your mark!
</p> <br> <br>
<h2 class="mb-4" style="font-weight: bold;">ABOUT IDEATHON</h2>
<p class="mb-4">
    Almost a year back we initiated the launch of PATTARAI with an Ideathon for our freshers. Since then
    we
    have come a long way... Technical events play a major role in shaping the minds of engineering
    students
    and giving them a great platform. We have been providing a lot of great opportunities for the students
    from licet, we felt now can be the right time to have an inter-college Ideathon. The topic we have
    chosen for this yearâ??s Ideathon is â??Reinventing Imaginationâ??. We are hoping to receive ideas that are
    realistic and also create a warping change that leaves an impact. We aim to break boundaries and the
    technical stigma that students assume.
</p> <br>
<h2 class="mb-4 font-weight-bold" style="font-weight: bold;">DATES ARE EXTENDED</h2>
<p>Due to large number of requests from the applicants, we have extended the submission deadline for
Ideathon#2.
    We are glad to inform you that the dates for the registration and submission of your abstract has been
    extended till 2nd of
    May - 12.00 am. So, make use of this time wisely and prepare. </p>
</div>

<div class="circle intro-circle-1"></div>
<div class="circle intro-circle-2"></div>
<div class="circle intro-circle-3"></div>
<div class="circle intro-circle-4"></div>
</div>
<!-- <div id="twitch-embed">

</div> -->

<!--Ideathon Prelims-->
<!-- <div class="position-relative tm-border-bottom intro-container mb-2">
    <div class="section dark ">
        <div class="header-section p-4">
            <h2 class="text-header white"><b>Prelims - </b><span class="font-weight-light">May 8, 2021</span></h2>
            <div class="divider center color"></div>
        </div>

        <div class="container1">
            <div role="list" class="w-dyn-items w-row">

                <div role="listitem" class="w-dyn-item w-col w-col-6">
                    <div class="blog-link">
                        <h4 class="blog-date">POOL A</h4>
                        <p class="post-paragraph light">2.00 pm - 5.00 pm</p>
                        <br>
                        <a href="https://meet.google.com/ygo-rqmx-dzh" aria-current="page"
                            class="button w-button w--current" target="top">Join Google Meet</a>
                    </div>
                </div>
            </div>
        </div>
    </div>
-->

```

```

    </div>
  </div>

  <div role="listitem" class="w-dyn-item w-col w-col-6">
    <div class="blog-link">
      <h4 class="blog-date">P00L B</h4>
      <p class="post-paragraph light">2.00 pm - 5.00 pm</p>
      <br>
      <a href="https://meet.google.com/ptj-dskt-cuv" aria-current="page"
        class="button w-button w--current" target="top">Join Google Meet</a>
    </div>
  </div>

  <div role="listitem" class="w-dyn-item w-col w-col-6">
    <div class="blog-link">
      <h4 class="blog-date">P00L C</h4>
      <p class="post-paragraph light">2.00 pm - 5.00 pm</p>
      <br>
      <a href="https://meet.google.com/aas-onmw-zkr" aria-current="page"
        class="button w-button w--current" target="top">Join Google Meet</a>
    </div>
  </div>

  <div role="listitem" class="w-dyn-item w-col w-col-6">
    <div class="blog-link">
      <h4 class="blog-date">P00L D</h4>
      <p class="post-paragraph light">2.00 pm - 5.00 pm</p>
      <br>
      <a href="https://meet.google.com/ary-bdrq-ymt" aria-current="page"
        class="button w-button w--current" target="top">Join Google Meet</a>
    </div>
  </div>

</div>

</div>
<div class="circle intro-circle-3"></div>
<div class="circle intro-circle-4"></div>
</div>
</div> -->

<!--Finals-->
<div class="position-relative tm-border-bottom intro-container mb-2">
  <div class="header-section p-4">
    <h2 class="text-header white"><b>Finals - </b><span class="font-weight-light">May 9, 2021</span></h2>
    <div class="divider center color"></div>
  </div>
  <div class="row justify-content-center">
    <div role="row listitem" class="px-4 py-2 w-dyn-item w-col w-col-12">
      <div class="blog-link">

```



```

        <h4 class="blog-date">FINALS</h4>
        <p class="post-paragraph light">The countdown is on!<br>
        Get ready for the Final Verdict.</p>
        <br>
        <a href="https://zoom.us/j/99393448111" aria-current="page"
        class="button w-button w--current" target="top">Join Zoom Meet</a>
    </div>
</div>

    <div class="circle intro-circle-3"></div>
    <div class="circle intro-circle-4"></div>
</div>

</div>
<!--Finals End-->

<div class="row justify-content-center">
    <div class="col-md-4 py-2 justify-content-center">
        

    </div>
    <!-- <div class="col-md-4 py-2 justify-content-center">
        
    </div> -->
    <div class="col-md-4 py-2 justify-content-center">
        <!--  -->
        
    </div>
    <div class="col-md-4 py-2 justify-content-center">
        
    </div>
</div>
</div>
</li>
<li data-page-no="2">
    <!-- Image Carousel -->
    <div class="mx-auto position-relative ">
        <div class="circle intro-circle-1"></div>
        <div class="circle intro-circle-2"></div>
        <div class="mx-auto tm-bg-dark pb-5 tm-border-top px-5">
            <h3 class="font-weight-bold" style="font-weight: bold; color: #FFC284;"> GUIDELINES</h3><br>
            <div class="pb-5">
                <h5 style="text-transform: uppercase;">Ideathon#2 is open for all schools and colleges.</h5>
                <p>We have few guidelines to be followed by the participants.<br> Please make sure that you fit to these
                requirements to participate in this event.</p>
            </div>
            <h5 class="font-weight-bold">TEAM SIZE</h5>
            A team can consists of minimum one and maximum of four members.
            <br><br><br>
            <h5 class="font-weight-bold">TEAM MEMBERS</h5>

```

```

A team can comprise members from any department within the college or the school.<br>
Inter-College or Inter-School teams will not encouraged.
<br><br><br>
<h5 class="font-weight-bold">PLAGIARISM</h5>
Plagiarism will not be tolerated. You can make changes and improvise to an already existing idea
<br><br><br>
<h5 class="font-weight-bold">NUMBER OF IDEAS</h5>
You are allowed to submit more than one idea.
</div>
</div>
</li>
<li data-page-no="3" class="px-3">
<div class="position-relative tm-border-top tm-border-bottom">
<div class="circle intro-circle-1"></div>
<div class="circle intro-circle-2"></div>
<div class="circle intro-circle-3"></div>
<div class="circle intro-circle-4"></div>
<div class="tm-bg-dark content-pad">
<h1 class="mb-5" style="font-weight: bold; color: #FFC284;">TRACKS</h1>
<h2> Bending reality with VR AR MR and AI </h2>
Here is the tech- stack of the future. Use your idea engines to come up with solutions to solve problems
faced by the common man. Let say that the equipment and materials used to make an AR based device or an
application is expensive, innovate a way to find alternatives or bring down the cost. Many of the
smartphones do not possess the capability of feeling the external camera in real time come up with an idea
to make this possible. The problem statements are not limited to these example you can come up with your
own statements and innovating the existing technology
<br><br>

<h2>Bridging the technological gap in rural areas</h2>

The issue of the â??digital divideâ??. the technological gap between the â??havesâ?? and the â??have notsâ??. has been
identified as a potential contributor to the widening income gap both within and among countries for about
20 years now.find ways to fix existing solutions which have their their fair contribution of disadvantages
for example
Connecting farmers to the global market or in-expensive ways to introduce drones or Iot in agriculture. As
this topic suggests you must focus on creating a tech ecosystem which benefits the rural side.The problem
statements are not limited to these example you can come up with your own Pain points or innovating the
existing technology

<br><br>

<h2>Robotics the future</h2>

As we all are moving towards the growth of technology. There are many innovations and inventions that will
change the future. But there are a lot of problems when it comes to implementation and the cost. So find
ways of implementing or making improvement in the existing innovations. You are allowed to bring your own
idea

<br><br>
<h2>Innovations on renewable energy</h2>

```

```

Carbon dioxide is in the air!!!. Come up with solutions to make renewable energy more efficient. Even
making a small change can impact the phase of the earth. Let's take a solar power plan, the efficiency has
not reached its full capacity. Do you know renewable energy still generates pollution you can find ways
controlling this factor
The problem statements are not limited to these example you can come up with your own Pain points or
innovating the existing technology

<br><br>
<h2>Innovation free for all</h2>

We don't want you to stick to a certain topic or domain. You can bring out your own ideas and
innovation. At the end of the day it must be your original idea or improved from an existing one. It can
solve a problem locally or a global crisis
happy thinking!!!!

</div>
<div class="mt-5 px-4">
  <h4 class="mb-3">Resources</h4>
  
  <!-- The Modal -->
  <div id="myModal" class="modal">
    <span class="close">&times;</span>
    <div class="modal-content" id="img01">
      <img id="caption"></div>
    </div>
  </div>
</div>
</li>
<li data-page-no="4">
  <div class="mx-auto page-width-3">
    <div class="row pb-2">
      <div class="col-md-6 tm-contact-right pt-5">
        <a target="_blank" href="#">
          <button class="btn bg-dark text-white mb-2">
            Registrations Closed!
          </button>
        </a>
      <br><br>
      <div class="bg-dark p-4">
        <p class="mb-4">
          Loyola Campus,
          Nungambakkam, Chennai
        </p>
      </div>
      <div>
        Mail Us: <a href="/cdn-cgi/l/email-protection#e6968792928794878fa88f858392c88785c88f88" class="tm-link-white"><span class="__cf_email__" data-cfemail="4e262b2222210e3e2f3a3a2f3c2f27602720">
{email&#160;protected}</span></a>
      </div>
      <div class="tm-mb-45 mt-2">
        Tel: <a href="tel:+91 79045 57742" class="tm-link-white">+91 79045 57742</a><br>
        Tel: <a href="tel:+91 86102 98756" class="tm-link-white">+91 86102 98756</a>
      </div>
    </div>
  </div>

```

```

    </div>
  </div>
  <!-- Map -->
  <div class="map-outer">
    <div class="gmap-canvas">
      <iframe width="100%" height="400" id="gmap-canvas"
        src="https://maps.google.com/maps?q=Licet%20pattarai&t=&z=13&ie=UTF8&iwloc=&output=embed"
        frameborder="0" scrolling="no" marginheight="0" marginwidth="0"></iframe>
    </div>
  </div>
  <div class="col-md-6">
    
  </div>
</div>
</div>
</li>
</ul>
</div>
<div class="container-fluid">
  <footer class="row mx-auto tm-footer">
    <div class="col-md-6 px-0">
      Copyright 2021 - Pattarai - IDEATHON. All rights reserved.
    </div>
    <div class="col-md-6 px-0 tm-footer-right">
      Designed by <a rel="sponsored" href="https://pattarai.in" target="_blank" class="tm-link-white">Pattarai</a>
    </div>
  </footer>
</div>
</div>
<!-- Preloader, https://ihatetomatoes.net/create-custom-preloading-screen/ -->
<div id="loader-wrapper">

  <div id="loader"></div>

  <div class="loader-section section-left">
  </div>
  <div class="loader-section section-right"></div>

```

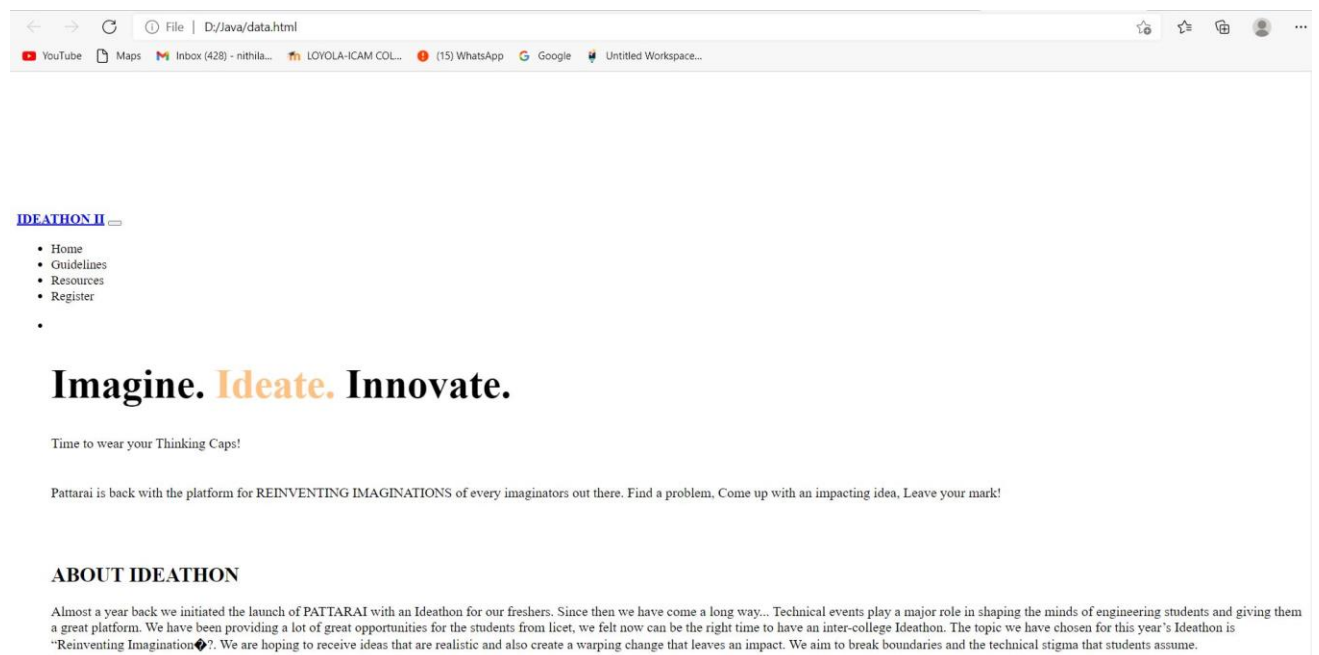
```

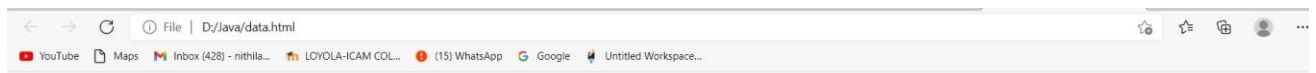
</div>
<script data-cfasync="false" src="/cdn-cgi/scripts/5c5dd728/cloudflare-static/email-decode.min.js"></script><script src="js/jquery-3.5.1.min.js"></script>
<script src="js/bootstrap.min.js"></script>
<script src="js/slick.js"></script>
<script src="js/templatemo-script.js"></script>
<script src="js/modal.js"></script>
<!-- Load the Twitch embed JavaScript file -->
<!-- <script src="https://embed.twitch.tv/embed/v1.js"></script> -->

<!-- Create a Twitch.Embed object that will render within the "twitch-embed" element -->
<!-- <script type="text/javascript">
  new Twitch.Embed("twitch-embed", {
    width: "100%",
    height: 800,
    channel: "licetpattarai",
    // Only needed if this page is going to be embedded on other websites
    parent: ["embed.example.com", "othersite.example.com"]
  });
</script> -->
</body>
</html>

```

Output - Client Side:





DATES ARE EXTENDED

Due to large number of requests from the applicants, we have extended the submission deadline for Ideathon#2. We are glad to inform you that the dates for the registration and submission of your abstract has been extended till 2nd of May - 12.00 am. So, make use of this time wisely and prepare.

Finals - May 9, 2021

FINALS

The countdown is on!
Get ready for the Final Verdict.

[Join Zoom Meet](#)







• GUIDELINES

IDEATHON#2 IS OPEN FOR ALL SCHOOLS AND COLLEGES.

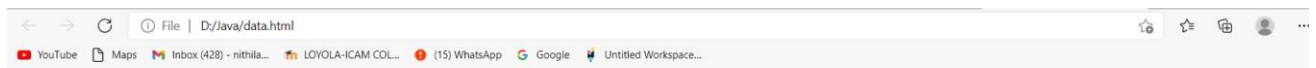
We have few guidelines to be followed by the participants.
Please make sure that you fit to these requirements to participate in this event.

TEAM SIZE

A team can consists of minimum one and maximum of four members.

TEAM MEMBERS

A team can comprise members from any department within the college or the school.



TEAM MEMBERS

A team can comprise members from any department within the college or the school.
Inter-College or Inter-School teams will not encouraged.

PLAGIARISM

Plagiarism will not be tolerated. You can make changes and improvise to an already existing idea

NUMBER OF IDEAS

You are allowed to submit more than one idea.

• TRACKS

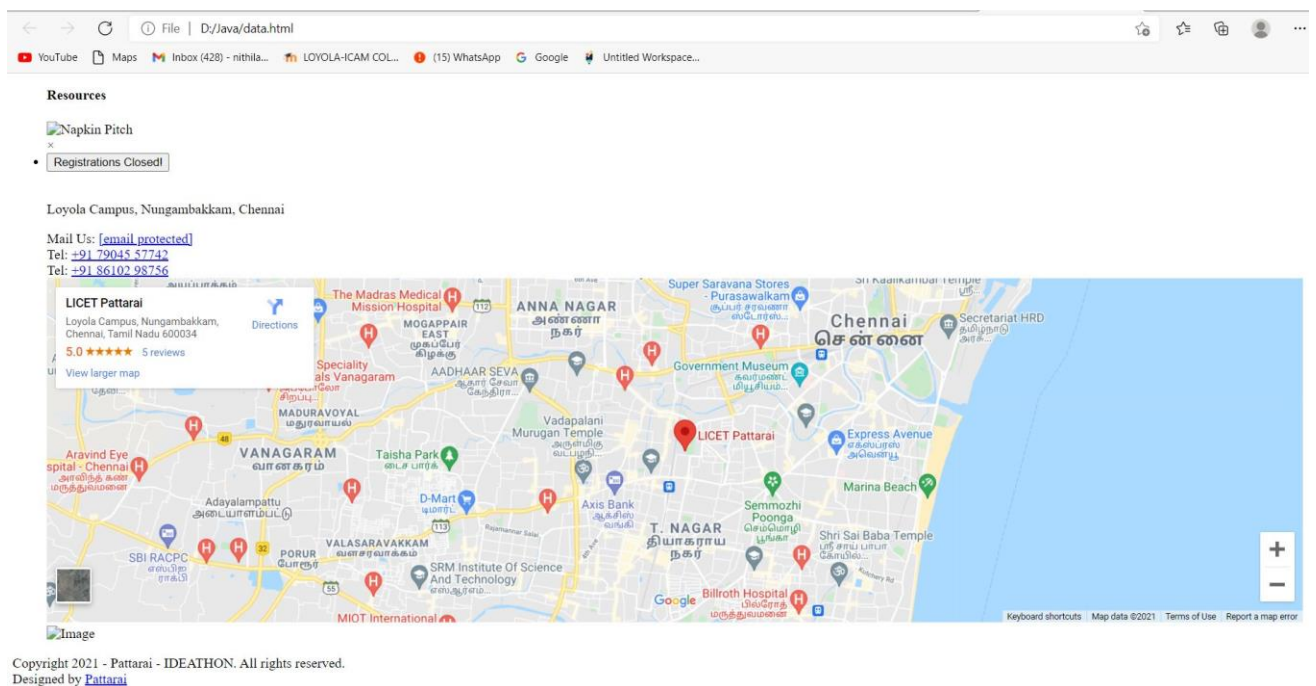
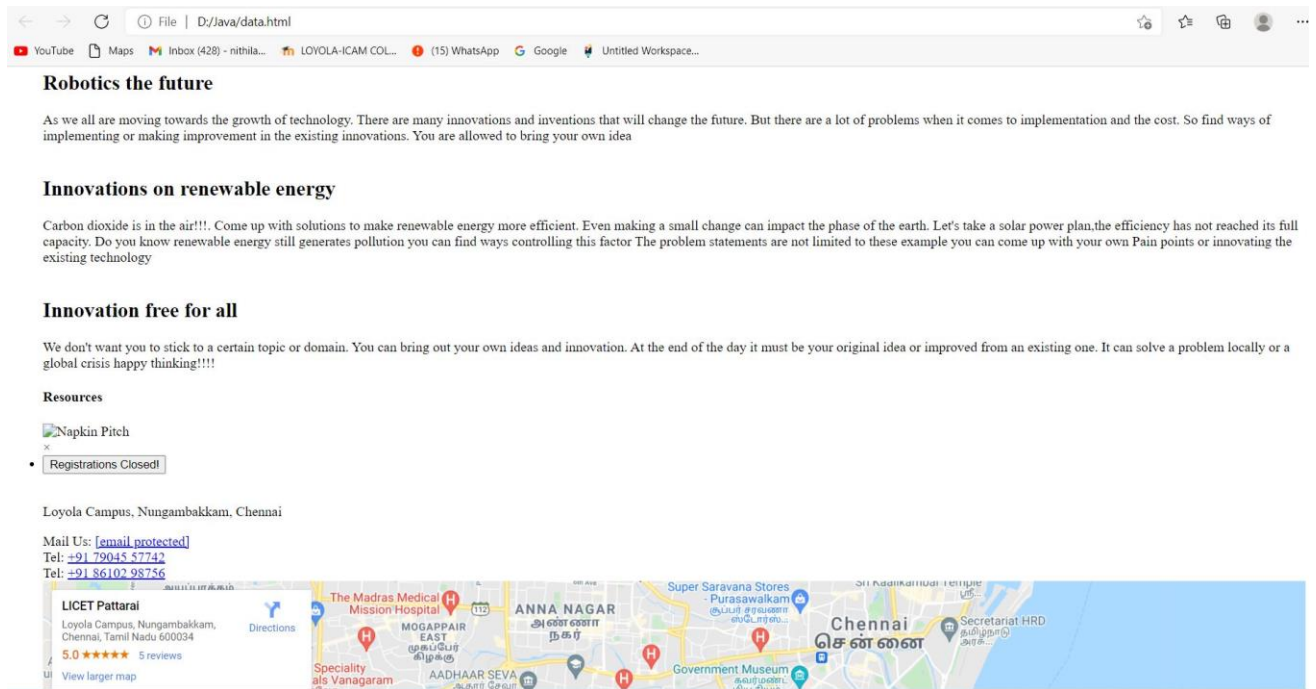
Bending reality with VR AR MR and AI

Here is the tech- stack of the future. Use your idea engines to come up with solutions to solve problems faced by the common man. Let say that the equipment and materials used to make an AR based device or an application is expensive, innovate a way to find alternatives or bring down the cost. Many of the smartphones do not possess the capability of feeling the external camera in real time come up with an idea to make this possible. The problem statements are not limited to these example you can come up with your own statements and innovating the existing technology

Bridging the technological gap in rural areas

The issue of the “digital divide”, the technological gap between the “haves” and the “have nots”, has been identified as a potential contributor to the widening income gap both within and among countries for about 20 years now.find ways to fix existing solutions which have their fair contribution of disadvantages for example Connecting farmers to the global market or in-expensive ways to introduce drones or IoT in agriculture. As this topic suggests you must focus on creating a tech ecosystem which benefits the rural side.The problem statements are not limited to these example you can come up with your own Pain points or innovating the existing technology

Robotics the future



Result:

Thus, the webpage is downloaded and both the output of server and client sides after executing have been attached here.

Aim:

To write a socket program for implementation of echo.

Algorithm:**Client Side**

1. Start the program.
2. Create a socket which binds the Ip address of server and the port address to acquire service.
3. After establishing connection send a data to server.
4. Receive and print the same data from server.
5. Close the socket.
6. End the program.

Server Side

1. Start the program.
2. Create a server socket to activate the port address.
3. Create a socket for the server socket which accepts the connection.
4. After establishing connection receive the data from client.
5. Print and send the same data to client.
6. Close the socket.
7. End the program.

Program:**Echo Client:**

```
import java.io.*;
import java.net.*;
public class eclient
{
    public static void main(String args[])
    {
        Socket c=null;
        String line;
        DataInputStream is,is1;
        PrintStream os;

        try {
            c=new Socket("localhost",8080);
        }

        catch(IOException e)
        {
            System.out.println(e);
        }
    }
}
```

```

try {
    os=new PrintStream(c.getOutputStream());
    is=new DataInputStream(System.in);
    is1=new DataInputStream(c.getInputStream());
    do
        {
            System.out.println("client");
            line=is.readLine();
            os.println(line);
            if(!line.equals("exit"))
                System.out.println("server:"+is1.readLine());
        }while(!line.equals("exit"));
    }
    catch(IOException e)
    {
        System.out.println("socket closed");
    }
}
}

```

Echo Server:

```

import java.io.*;
import java.net.*;
import java.lang.*;

public class eserver
{
    public static void main(String args[])throws IOException
    {
        ServerSocket s=null;
        String line;
        DataInputStream is;
        PrintStream ps;
        Socket c=null;
        try {
            s=new ServerSocket(8080);
        }
        catch(IOException e)
        {
            System.out.println(e);
        }
        try {
            c=s.accept();
            is=new DataInputStream(c.getInputStream());
            ps=new PrintStream(c.getOutputStream());
            while(true) {
                line=is.readLine();
                System.out.println("msg received and sent back to client");
            }
        }
    }
}

```

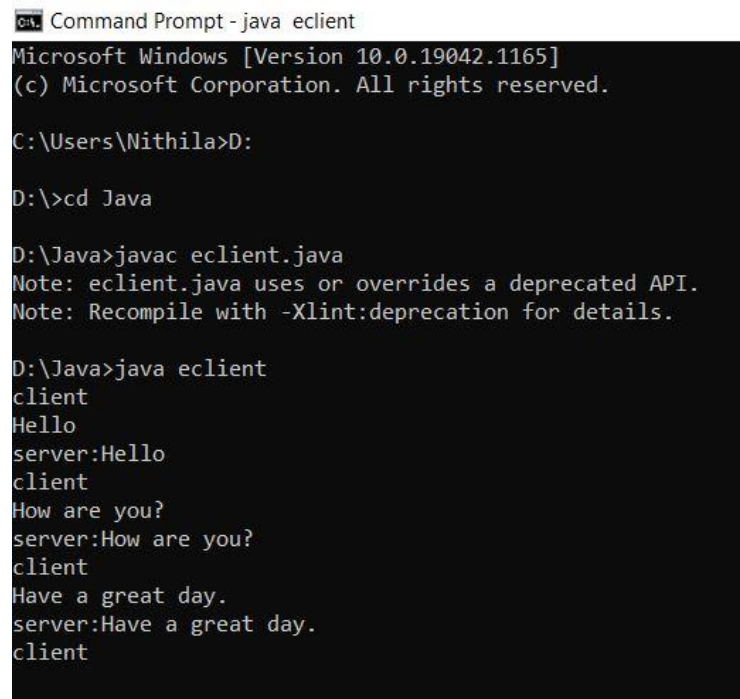
```

        ps.println(line);
    }
}
catch(IOException e)
{
    System.out.println(e);
}
}
}

```

Output:

Client



```

Command Prompt - java eclient
Microsoft Windows [Version 10.0.19042.1165]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Nithila>D:

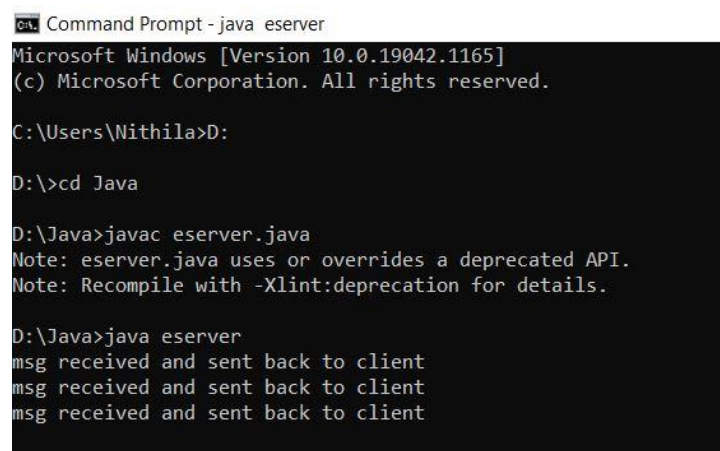
D:\>cd Java

D:\Java>javac eclient.java
Note: eclient.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.

D:\Java>java eclient
client
Hello
server:Hello
client
How are you?
server:How are you?
client
Have a great day.
server:Have a great day.
client

```

Server



```

Command Prompt - java eserver
Microsoft Windows [Version 10.0.19042.1165]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Nithila>D:

D:\>cd Java

D:\Java>javac eserver.java
Note: eserver.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.

D:\Java>java eserver
msg received and sent back to client
msg received and sent back to client
msg received and sent back to client

```

Result:

Thus, the program for simulation of echo server was written & executed.

Aim:

To write client-server application for chat using TCP.

Algorithm:**Client**

1. Start the program
2. Include necessary package in java
3. To create a socket in client to server
4. The client establishes a connection to the server
5. The client accepts the connection and to send the data from client to server
6. The client communicates to the server to send the end of the message
7. Stop the program.

Server

1. Start the program
2. Include necessary package in java
3. To create a socket in server to client
4. The server establishes a connection to client
5. The server accepts the connection and to send the data from server to client and
6. vice versa
7. The server communicates to the client to send the end of the message
8. Stop the program.

Program:**TCPserver1.java**

```
import java.net.*;
import java.io.*;
public class TCPserver1
{
    public static void main( String args[]) throws Exception
    {
        ServerSocket srs = new ServerSocket(1234);
        System.out.println("Server is running...");
        Socket ss=srs.accept();
        System.out.println("connection establised");
        BufferedReader kb = new BufferedReader(new InputStreamReader(System.in));
        BufferedReader br = new BufferedReader(new
InputStreamReader(ss.getInputStream()));
        DataOutputStream dos = new DataOutputStream(ss.getOutputStream());

        while(true)
        {
            //System.out.println("server repeat as long as client not send null");
            String s2,s3;
            while((s2=br.readLine())!=null)
```

```

        {
            System.out.println("Client said : "+s2);
            System.out.println("Enter text ");
            s3 = kb.readLine();
            //System.out.println("Answer send to client machine");
            dos.writeBytes(s3+"\n");
        }
        System.out.println("Terminated..");
        ss.close();
        srs.close();
        dos.close();
        kb.close();
        System.exit(0);
    }
}

```

TCPclient1.java

```

import java.net.*;
import java.io.*;
public class TCPclient1
{
    public static void main( String args[]) throws Exception
    {
        Socket cs = new Socket("localhost",1234);
        BufferedReader kb = new BufferedReader(new InputStreamReader(System.in));
        BufferedReader br = new BufferedReader(new
InputStreamReader(cs.getInputStream()));
        DataOutputStream dos = new DataOutputStream(cs.getOutputStream());
        System.out.println(" Enter text..");
        System.out.println(" if client 'quit' type  exit");
        String s1,s4=null;
        while(!(s1=kb.readLine()).equals("exit"))
        {
            //System.out.println(" data  send to server machine");
            dos.writeBytes(s1+"\n");
            s4 = br.readLine();
            //System.out.println(" data  receive from  server machine");
            System.out.println("Server said : "+s4);
            System.out.println("Enter text ");
        }
        System.out.println("Terminated..");
        cs.close();
        dos.close();
        kb.close();
    }
}

```

Output:

Client

```
Command Prompt
Microsoft Windows [Version 10.0.19042.1165]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Nithila>D:

D:\>cd Java

D:\Java>javac TCPclient1.java

D:\Java>java TCPclient1
Enter text..
if client 'quit' type  exit
Hi
Server said : Hello
Enter text
How are you?
Server said : I'm fine
Enter text
Have a great day!
Server said : Bye
Enter text
exit
Terminated..

D:\Java>
```

Server

```
Command Prompt
Microsoft Windows [Version 10.0.19042.1165]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Nithila>D:

D:\>cd Java

D:\Java>javac TCPserver1.java

D:\Java>java TCPserver1
Server is running...
connection established
Client said : Hi
Enter text
Hello
Client said : How are you?
Enter text
I'm fine
Client said : Have a great day!
Enter text
Bye
Terminated..

D:\Java>
```

Result:

Thus, the program for simulation of client-server application for chat using TCP was written & executed.

Aim:

To Perform File Transfer in Client & Server Using TCP/IP.

Algorithm:**Client Side**

1. Start.
2. Establish a connection between the Client and Server.
3. Socket ss=new Socket(InetAddress.getLocalHost(),1100);
4. Implement a client that can send two requests.
 - i) To get a file from the server.
 - ii) To put or send a file to the server.
5. After getting approval from the server ,the client either get file from the server or send
6. file to the server.

Server Side

1. Start.
2. Implement a server socket that listens to a particular port number.
3. Server reads the filename and sends the data stored in the file for the 'get' request.
4. It reads the data from the input stream and writes it to a file in the server for the 'put' instruction.
5. Exit upon client's request.
6. Stop.

Program:**Client side**

```
import java.io.BufferedOutputStream;
import java.io.FileOutputStream;
import java.io.InputStream;
import java.net.InetAddress;
import java.net.Socket;
```

```
public class FileClient {
```

```
    public static void main(String[] args) throws Exception{
```

```
        //Initialize socket
```

```
        Socket socket = new Socket(InetAddress.getByName("localhost"), 5000);
        byte[] contents = new byte[10000];
```

```
        //Initialize the FileOutputStream to the output file's full path.
```

```
        FileOutputStream fos = new FileOutputStream("D:\\Java\\data2.txt");
        BufferedOutputStream bos = new BufferedOutputStream(fos);
        InputStream is = socket.getInputStream();
```



```

//No of bytes read in one read() call
int bytesRead = 0;

while((bytesRead=is.read(contents))!=-1)
    bos.write(contents, 0, bytesRead);

bos.flush();
socket.close();

System.out.println("File saved successfully!");
}
}

```

Server Side

```

import java.io.BufferedInputStream;
import java.io.File;
import java.io.FileInputStream;
import java.io.OutputStream;
import java.net.InetAddress;
import java.net.ServerSocket;
import java.net.Socket;

public class FileServer {

    public static void main(String[] args) throws Exception {
        //Initialize Sockets
        ServerSocket ssock = new ServerSocket(5000);
        Socket socket = ssock.accept();

        //The InetAddress specification
        InetAddress IA = InetAddress.getByName("localhost");

        //Specify the file
        File file = new File("D:\\Java\\data1.txt");
        FileInputStream fis = new FileInputStream(file);
        BufferedInputStream bis = new BufferedInputStream(fis);
        //Get socket's output stream
        OutputStream os = socket.getOutputStream();

        //Read File Contents into contents array
        byte[] contents;
        long fileLength = file.length();
        long current = 0;
        long start = System.nanoTime();
        while(current!=fileLength){
            int size = 10000;
            if(fileLength - current >= size)
                current += size;

```

```

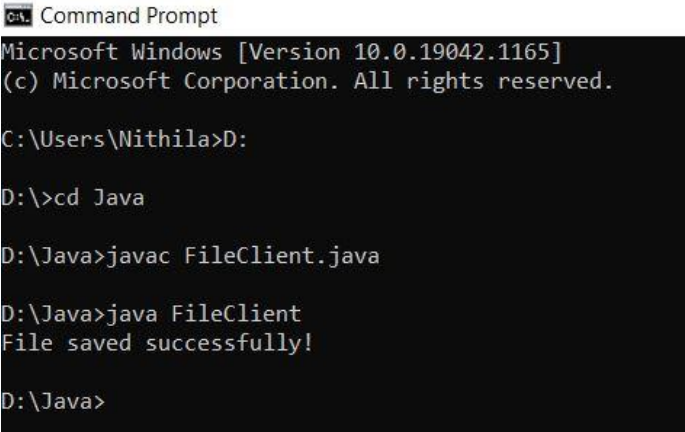
        else{
            size = (int)(fileLength - current);
            current = fileLength;
        }
        contents = new byte[size];
        bis.read(contents, 0, size);
        os.write(contents);
        System.out.print("Sending file ... "+(current*100)/fileLength+"% complete!");
    }

    os.flush();
    //File transfer done. Close the socket connection!
    socket.close();
    ssock.close();
    System.out.println("File sent succesfully!");
}
}

```

Output:

Client



```

C:\> Command Prompt
Microsoft Windows [Version 10.0.19042.1165]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Nithila>D:

D:\>cd Java

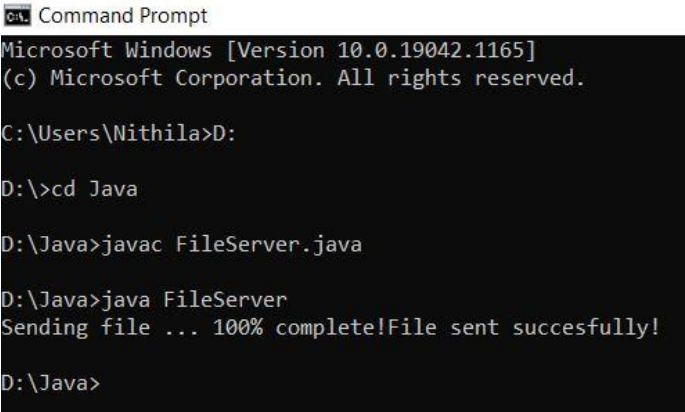
D:\Java>javac FileClient.java

D:\Java>java FileClient
File saved successfully!

D:\Java>

```

Server



```

C:\> Command Prompt
Microsoft Windows [Version 10.0.19042.1165]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Nithila>D:

D:\>cd Java

D:\Java>javac FileServer.java

D:\Java>java FileServer
Sending file ... 100% complete!File sent succesfully!

D:\Java>

```

Result:

Thus, the program for performing file transfer in client & server using TCP/IP was written & executed.

Aim:

To perform domain name system using UDP.

Client:

```
import java.net.*;
import java.util.*;
class Clientdns12
{
    public static void main(String args[])
    {
        try
        {
            DatagramSocket client=new DatagramSocket();
            InetAddress addr=InetAddress.getByName("127.0.0.1");

            byte[] sendbyte=new byte[1024];
            byte[] receivebyte=new byte[1024];
            BufferedReader in=new BufferedReader(new
InputStreamReader(System.in));
            System.out.println("Enter the DOMAIN NAME or IP address:");
            String str=in.readLine();
            sendbyte=str.getBytes();
            DatagramPacket sender=new
DatagramPacket(sendbyte,sendbyte.length,addr,1309);
            client.send(sender);
            DatagramPacket receiver=new
DatagramPacket(receivebyte,receivebyte.length);
            client.receive(receiver);
            String s=new String(receiver.getData());
            System.out.println("IP address or DOMAIN NAME: "+s.trim());
            client.close();
        }
        catch(Exception e)
        {
            System.out.println(e);
        }
    }
}
```

Server:

```
import java.io.*;
import java.net.*;
import java.util.*;
class Serverdns12
```

```

{
    public static void main(String args[])
    {
        try
        {
            DatagramSocket server=new DatagramSocket(1309);
            while(true)
            {
                byte[] sendbyte=new byte[1024];
                byte[] receivebyte=new byte[1024];
                DatagramPacket receiver=new
DatagramPacket(receivebyte,receivebyte.length);
                server.receive(receiver);
                String str=new String(receiver.getData());
                String s=str.trim();
                //System.out.println(s);
                InetAddress addr=receiver.getAddress();
                int port=receiver.getPort();
                String ip[]={"165.165.80.80","165.165.79.1"};
                String
name[]={"www.apptitudeguru.com", "www.downloadcyclone.blogspot.com"};
                for(int i=0;i<ip.length;i++)
                {
                    if(s.equals(ip[i]))
                    {
                        sendbyte=name[i].getBytes();
                        DatagramPacket sender=new
DatagramPacket(sendbyte,sendbyte.length,addr,port);
                        server.send(sender);
                        break;
                    }
                    else if(s.equals(name[i]))
                    {
                        sendbyte=ip[i].getBytes();
                        DatagramPacket sender=new
DatagramPacket(sendbyte,sendbyte.length,addr,port);
                        server.send(sender);
                        break;
                    }
                }
            }
        }
        catch(Exception e)
        {
            System.out.println(e);
        }
    }
}

```

```
}  
}
```

Output:

Client

```
C:\> Command Prompt  
Microsoft Windows [Version 10.0.19041.1237]  
(c) Microsoft Corporation. All rights reserved.  
  
C:\Users\ctadmin>cd Nithila  
  
C:\Users\ctadmin\Nithila>javac Clientdns12.java  
  
C:\Users\ctadmin\Nithila>java Clientdns12  
Enter the DOMAIN NAME or IP address:  
www.downloadcyclone.blogspot.com  
IP address or DOMAIN NAME: 165.165.79.1  
  
C:\Users\ctadmin\Nithila>
```

Server

```
C:\> Command Prompt  
Microsoft Windows [Version 10.0.19041.1237]  
(c) Microsoft Corporation. All rights reserved.  
  
C:\Users\ctadmin>cd Nithila  
  
C:\Users\ctadmin\Nithila>javac Serverdns12.java  
  
C:\Users\ctadmin\Nithila>java Serverdns12  
  
C:\Users\ctadmin\Nithila>
```

Result:

Thus, domain name system is performed using UDP.

EXP: 5A IMPLEMENTATION OF ADDRESS RESOLUTION PROTOCOL DATE: 24-09-2021

Aim:

To implement the Address Resolution Protocol.

Algorithm:

Client:

1. Start the program
2. Using socket connection is established between client and server.
3. Get the IP address to be converted into MAC address.
4. Send this IP address to server.
5. Server returns the MAC address to client.

Server:

1. Start the program
2. Accept the socket which is created by the client.
3. Server maintains the table in which IP and corresponding MAC addresses are stored.
4. Read the IP address which is send by the client.
5. Map the IP address with its MAC address and return the MAC address to client.

Code:

Arp Client Pgm:

```
import java.io.*;
import java.net.*;
import java.util.*;
class ArpClient
{
    public static void main(String args[])
    {
        try
        {
            BufferedReader in=new BufferedReader(new
InputStreamReader(System.in));
            Socket clsct=new Socket("127.0.0.1",5604);
            DataInputStream din=new DataInputStream(clsct.getInputStream());
            DataOutputStream dout=new
DataOutputStream(clsct.getOutputStream());
            System.out.println("Enter the Logical address(IP):");
            String str1=in.readLine();
            dout.writeBytes(str1+'\n');
            String str=din.readLine();
            System.out.println("The Physical Address is: "+str);
            clsct.close();
        }
        catch (Exception e)
```

```

        {
            System.out.println(e);
        }
    }
}

```

Arp Server:

```

import java.io.*;
import java.net.*;
import java.util.*;
class ArpServer
{
    public static void main(String args[])
    {
        try
        {
            ServerSocket obj=new
            ServerSocket(5604);
            Socket obj1=obj.accept();
            while(true)
            {
                DataInputStream din=new DataInputStream(obj1.getInputStream());
                DataOutputStream dout=new
                DataOutputStream(obj1.getOutputStream());
                String str=din.readLine();
                String ip[]={"165.165.80.80","165.165.79.1"};
                String mac[]={"6A:08:AA:C2","8A:BC:E3:FA"};
                for(int i=0;i<ip.length;i++)
                {
                    if(str.equals(ip[i]))
                    {
                        dout.writeBytes(mac[i]+'\\n');
                        break;
                    }
                }
                obj.close();
            }
        }
        catch(Exception e)
        {
            System.out.println(e);
        }
    }
}

```

Output:

Client

```
Command Prompt
Microsoft Windows [Version 10.0.19041.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ctadmin>cd Nithila

C:\Users\ctadmin\Nithila>javac ArpClient.java
Note: ArpClient.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.

C:\Users\ctadmin\Nithila>java ArpClient
Enter the Logical address(IP):
165.165.80.80
The Physical Address is: 6A:08:AA:C2

C:\Users\ctadmin\Nithila>
```

Server

```
Command Prompt
Microsoft Windows [Version 10.0.19041.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ctadmin>cd Nithila

C:\Users\ctadmin\Nithila>javac ArpServer.java
Note: ArpServer.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.

C:\Users\ctadmin\Nithila>java ArpServer
java.lang.NullPointerException: Cannot invoke "String.equals(Object)" because "<local5>" is null

C:\Users\ctadmin\Nithila>
```

Result:

Thus, the Address Resolution Protocol is implemented.

EXP: 5B IMPLEMENTATION OF REVERSE ADDRESS RESOLUTION PROTOCOL

DATE: 24-09-2021

Aim:

To implement the Reverse Address Resolution Protocol (RARP).

Algorithm:

Client:

1. Start the program
2. Using socket connection is established between client and server.
3. Get the MAC address to be converted into IP address.
4. Send this MAC address to server.
5. Server returns the IP address to client.

Server:

1. Start the program
2. Accept the socket which is created by the client.
3. Server maintains the table in which MAC and corresponding IP addresses are stored.
4. Read the MAC address which is send by the client.
5. Map the MAC address with its IP address and return the IP address to client.

Code:

RARP.java - Client

```
import java.io.*;
import java.net.*;
import java.util.*;

public class RARP {
    public static void main(String args[]){
        try{
            DatagramSocket client = new DatagramSocket();
            InetAddress addr = InetAddress.getByName("127.0.0.1");
            byte[] sendByte = new byte[1204];
            byte[] receiveByte = new byte[1024];
            BufferedReader in = new BufferedReader(new InputStreamReader(System.in));
            System.out.println("Enter the Physical Address ");
            String str = in.readLine();
            sendByte = str.getBytes();
            DatagramPacket sender = new DatagramPacket(sendByte,sendByte.length,addr,1309);
            client.send(sender);
            DatagramPacket receiver = new DatagramPacket(receiveByte,receiveByte.length);
            client.receive(receiver);
            String s = new String(receiver.getData());
            System.out.println("The Logical Address is :"+ s.trim());
            client.close();
        }
        catch(Exception e){
```

```

        System.out.println(e);
    }
}
}

```

RARPServer.java - Server

```

import java.io.*;
import java.net.*;
import java.util.*;

public class RARPServer{

    public static void main(String args[]) {
        try{

            DatagramSocket server = new DatagramSocket(1309);
            while(true){
                byte[] sendByte = new byte[1204];
                byte[] receiveByte = new byte[1204];
                DatagramPacket receiver = new DatagramPacket(receiveByte, receiveByte.length);
                server.receive(receiver);
                String str = new String(receiver.getData());
                String s = str.trim();
                InetAddress addr = receiver.getAddress();
                int port = receiver.getPort();
                String ip[] = {"165.165.80.80"};
                String mac[] = {"6A:08:AA:C2"};

                for (int i = 0; i < ip.length; i++) {
                    if(s.equals(mac[i]))
                    {
                        sendByte = ip[i].getBytes();
                        DatagramPacket sender = new
DatagramPacket(sendByte, sendByte.length, addr, port);
                        server.send(sender);
                        break;
                    }
                }
                break;
            }
        }catch(Exception e)
        {
            System.out.println(e);
        }
    }
}

```

Output:

Client

```
Command Prompt
Microsoft Windows [Version 10.0.19041.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ctadmin>cd Nithila

C:\Users\ctadmin\Nithila>javac RARP.java

C:\Users\ctadmin\Nithila>java RARP
Enter the Physical Address
6A:08:AA:C2
The Logical Address is :165.165.80.80

C:\Users\ctadmin\Nithila>
```

Server

```
Command Prompt
Microsoft Windows [Version 10.0.19041.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ctadmin>cd Nithila

C:\Users\ctadmin\Nithila>javac RARPServer.java

C:\Users\ctadmin\Nithila>java RARPServer

C:\Users\ctadmin\Nithila>
```

Result:

Thus, the Reverse Address Resolution Protocol is implemented.

EXP: 6 STUDY OF NETWORK SIMULATOR AND SIMULATION OF CONGESTION

CONTROL ALGORITHMS USING NS

DATE: 30-09-2021

Aim:

To Study of Network simulator (NS).and Simulation of Congestion Control Algorithms using NS
NET WORK SIMULATOR (NS2)

- Ns programming: A Quick start
- Case study I: A simple Wireless network
- Case study II: Create a new agent in Ns

Ns overview

- Ns Status
- Periodical release (ns-2.26, Feb 2003)
- Platform support
- FreeBSD, Linux, Solaris, Windows and Mac

Ns Functionalities

Routing, Transportation, Traffic sources, Queuing disciplines, QoS, Wireless Ad hoc routing, mobile IP, sensor-MAC Tracing, visualization and various utilities NS(Network Simulators)

Most of the commercial simulators are GUI driven, while some network simulators are CLI driven. The network model / configuration describes the state of the network (nodes,routers, switches, links) and the events (data transmissions, packet error etc.). An important output of simulations are the trace files. Trace files log every packet, every event that occurred in the simulation and are used for analysis. Network simulators can also provide other tools to facilitate visual analysis of trends and potential trouble spots.

Most network simulators use discrete event simulation, in which a list of pending "events" is stored, and those events are processed in order, with some events triggering future events—such as the event of the arrival of a packet at one node triggering the event of the arrival of that packet at a downstream node.

Simulation of networks is a very complex task. For example, if congestion is high, then estimation of the average occupancy is challenging because of high variance. To estimate the likelihood of a buffer overflow in a network, the time required for an accurate answer can be extremely large. Specialized techniques such as "control variates" and "importance sampling" have been developed to speed simulation.

Examples of network simulators

There are many both free/open-source and proprietary network simulators. Examples of notable network simulation software are, ordered after how often they are mentioned in research papers:

1. ns (open source)

2. OPNET (proprietary software)
3. NetSim (proprietary software)

Uses of network simulators

Network simulators serve a variety of needs. Compared to the cost and time involved in setting up an entire test bed containing multiple networked computers, routers and data links, network simulators are relatively fast and inexpensive. They allow engineers, researchers to test scenarios that might be particularly difficult or expensive to emulate using real hardware - for instance, simulating a scenario with several nodes or experimenting with a new protocol in the network. Network simulators are particularly useful in allowing researchers to test new networking protocols or changes to existing protocols in a controlled and reproducible environment. A typical network simulator encompasses a wide range of networking technologies and can help the users to build complex networks from basic building blocks such as a variety of nodes and links. With the help of simulators, one can design hierarchical networks using various types of nodes like computers, hubs, bridges, routers, switches, links, mobile units etc.

Various types of Wide Area Network (WAN) technologies like TCP, ATM, IP etc. and Local Area Network (LAN) technologies like Ethernet, token rings etc., can all be simulated with a typical simulator and the user can test, analyze various standard results apart from devising some novel protocol or strategy for routing etc. Network simulators are also widely used to simulate battlefield networks in Network-centric warfare

There are a wide variety of network simulators, ranging from the very simple to the very complex. Minimally, a network simulator must enable a user to represent a network topology, specifying the nodes on the network, the links between those nodes and the traffic between the nodes. More complicated systems may allow the user to specify everything about the protocols used to handle traffic in a network. Graphical applications allow users to easily visualize the workings of their simulated environment. Text-based applications may provide a less intuitive interface, but may permit more advanced forms of customization.

Packet loss occurs when one or more packets of data travelling across a computer network fail to reach their destination. Packet loss is distinguished as one of the three main error types encountered in digital communications; the other two being bit error and spurious packets caused due to noise.

Packets can be lost in a network because they may be dropped when a queue in the network node overflows. The amount of packet loss during the steady state is another important property of a congestion control scheme. The larger the value of packet loss, the more difficult it is for transport layer protocols to maintain high bandwidths, the sensitivity to loss of individual packets, as well as to frequency and patterns of loss among longer packet sequences is strongly dependent on the application itself.

Throughput

This is the main performance measure characteristic, and most widely used. In communication networks, such as Ethernet or packet radio, throughput or network throughput is the average rate of successful message delivery over a communication channel. The throughput is usually measured

in bits per second (bit/s or bps), and sometimes in data packets per second or data packets per time slot. This measure shows how soon the receiver is able to get a certain amount of data sent by the sender. It is determined as the ratio of the total data received to the end-to-end delay. Throughput is an important factor which directly impacts the network performance.

Delay

Delay is the time elapsed while a packet travels from one point e.g., source premise or network ingress to destination premise or network egress. The larger the value of delay, the more difficult it is for transport layer protocols to maintain high bandwidths. We will calculate end-to-end delay.

Queue Length

A queuing system in networks can be described as packets arriving for service, waiting for service if it is not immediate, and if having waited for service, leaving the system after being served. Thus, queue length is a very important characteristic to determine how well the active queue management of the congestion control algorithm has been working.

Result:

Thus, the study of Network simulator (NS) and Simulation of Congestion Control Algorithms using NS has been done successfully.

EXP:7 CREATION OF NODES USING TCP IN NETWORK SIMULATOR DATE: 28-09-2021

Aim:

To write a simulator program for creating nodes in TCP for network communication.

Procedure:

1. Start, Set a new simulator ns
2. Open a file tr in write mode and ftr to output file
3. Create 2 nodes for file transfer
4. Create a duplex link for 2-way transfer with 2 mb speed 4 ms time
5. Establish connection using agent for communication of two nodes
6. Open connection for data transfer
7. Set object ftp and open ftp
8. Create a procedure finish. declare globals
9. Identify declared variables using flush
10. Close and declare time for start, stop and finish.

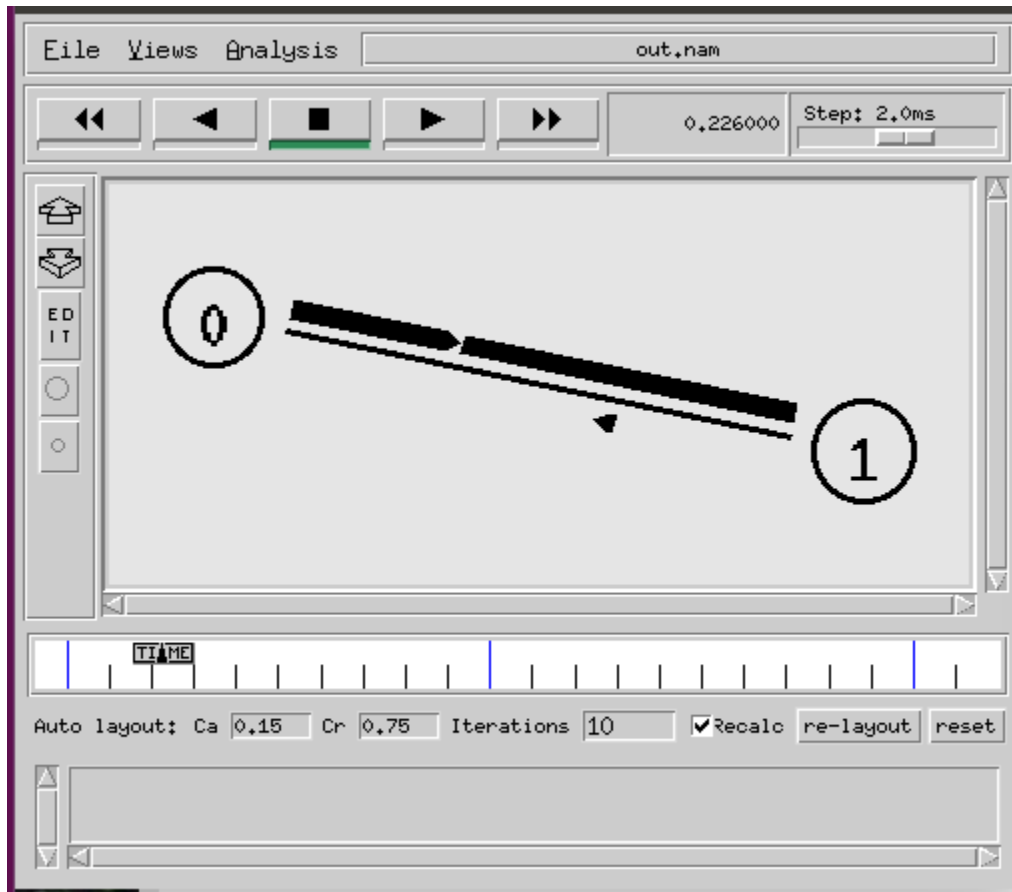
Code:

```
set ns [new Simulator]
set tr [open out.tr w]
$ns trace-all $tr
set ftr [open out.nam w]
$ns namtrace-all $ftr
set n0 [$ns node]
set n1 [$ns node]
$ns duplex-link $n0 $n1 2Mb 4ms DropTail
set tcp1 [new Agent/TCP]
set sink [new Agent/TCPSink]
$ns attach-agent $n0 $tcp1
$ns attach-agent $n1 $sink
$ns connect $tcp1 $sink
set ftp [new Application/FTP]
$ftp attach-agent $tcp1
proc finish {} {
    global ns tr ftr
    $ns flush-trace
    close $tr
    close $ftr
    exec nam out.nam &
    exit 0
}
```



```
}  
$ns at .1 "$ftp start"  
$ns at 2.0 "$ftp stop"  
$ns at 2.1 "finish"  
$ns run
```

Output:



Result:

Thus, the Simulation Program to Create nodes using TCP in a network simulator tool is Executed and the output is verified.

EXP: 8A NS2 SIMULATION USING DISTANCE VECTOR ROUTING PROTOCOL

DATE: 30.09.2021

Aim:

To implement NS2 simulation using Distance Vector Routing protocol.

Procedure:

1. Create a simulator object
2. Set routing protocol to Distance Vector routing
3. Trace packets on all links onto NAM trace and text trace file
4. Define finish procedure to close files, flush tracing and run NAM
5. Create four nodes
6. Specify the link characteristics between nodes
7. Add UDP agent for node n2
8. Create CBR traffic on top of UDP and set traffic parameters.
9. Add a sink agent to node n3
10. Connect source and the sink
11. Schedule events as follows:
 - a. Start traffic flow at 0.0
 - b. Call finish procedure at 5.0
12. Start the scheduler

Code:

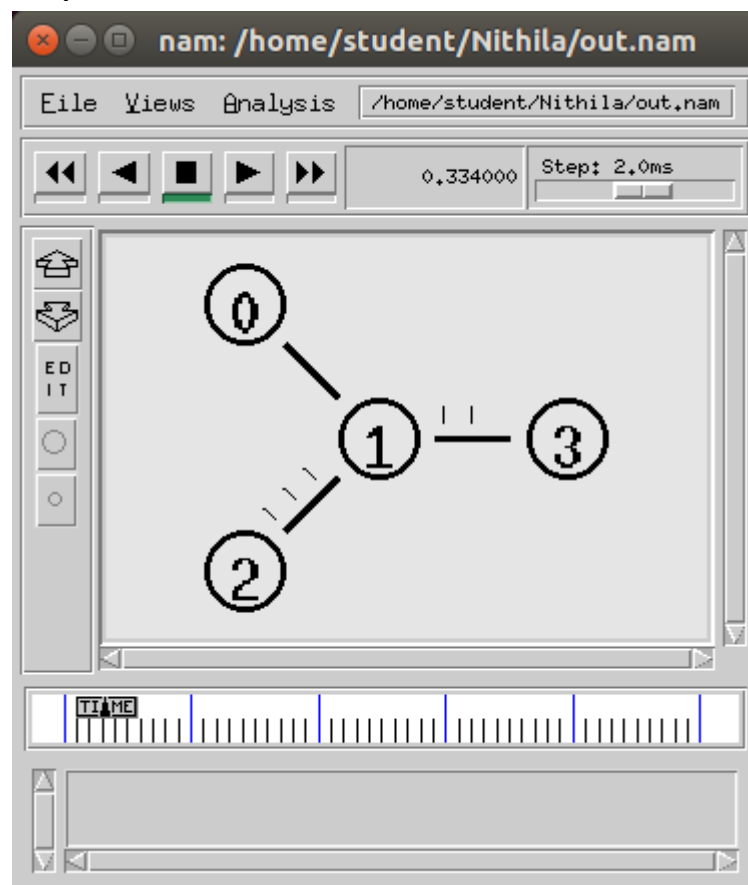
```
set ns [new Simulator]
set nf [open out.nam w]
$ns namtrace-all $nf
set tr [open out.tr w]
$ns trace-all $tr
proc finish {} {
    global nf ns tr
    $ns flush-trace
    close $tr
    exec nam out.nam &
    exit 0
}
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
$ns duplex-link $n0 $n1 10Mb 10ms DropTail
$ns duplex-link $n1 $n3 10Mb 10ms DropTail
$ns duplex-link $n2 $n1 10Mb 10ms DropTail
$ns duplex-link-op $n0 $n1 orient right-down
$ns duplex-link-op $n1 $n3 orient right
$ns duplex-link-op $n2 $n1 orient right-up
set tcp [new Agent/TCP]
```

```

$ns attach-agent $n0 $tcp
set ftp [new Application/FTP]
$ftp attach-agent $tcp
set sink [new Agent/TCPSink]
$ns attach-agent $n3 $sink
set udp [new Agent/UDP]
$ns attach-agent $n2 $udp
set cbr [new Application/Traffic/CBR]
$cbr attach-agent $udp
set null [new Agent/Null]
$ns attach-agent $n3 $null
$ns connect $tcp $sink
$ns connect $udp $null
$ns rtmodel-at 1.0 down $n1 $n3
$ns rtmodel-at 2.0 up $n1 $n3
$ns rproto DV
$ns at 0.0 "$ftp start"
$ns at 0.0 "$cbr start"
$ns at 5.0 "finish"
$ns run

```

Output:



Result:

Thus, NS2 simulation using Distance Vector Routing protocol has been implemented and executed successfully.

EXP:8B NS2 SIMULATION USING LINK STATE ROUTING PROTOCOL DATE: 30-09-2021

Aim:

To implement NS2 simulation using Link state routing protocol.

Procedure:

1. Create a Simulator object.
2. Set routing as dynamic.
3. Open the trace and nam trace files.
4. Define the finish procedure.
5. Create nodes and the links between them.
6. Create the agents and attach them to the nodes.
7. Create the applications and attach them to the udp agent.
8. Connect udp and null..
9. At 1 sec the link between node 1 and 2 is broken.
10. At 2 sec the link is up again.
11. Run the simulation.

Code:

```
set ns [new Simulator]
set nf [open out.nam w]
$ns namtrace-all $nf

set tr [open out.tr w]
$ns trace-all $tr

proc finish {} {
    global nf ns tr
    $ns flush-trace
    close $tr
    exec nam out.nam &
    exit 0
}

set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]

$ns duplex-link $n0 $n1 10Mb 10ms DropTail
$ns duplex-link $n1 $n3 10Mb 10ms DropTail
$ns duplex-link $n2 $n1 10Mb 10ms DropTail
```

```
$ns duplex-link-op $n0 $n1 orient right-down
$ns duplex-link-op $n1 $n3 orient right
$ns duplex-link-op $n2 $n1 orient right-up
```

```
set tcp [new Agent/TCP]
$ns attach-agent $n0 $tcp
```

```
set ftp [new Application/FTP]
$ftp attach-agent $tcp
```

```
set sink [new Agent/TCPSink]
$ns attach-agent $n3 $sink
```

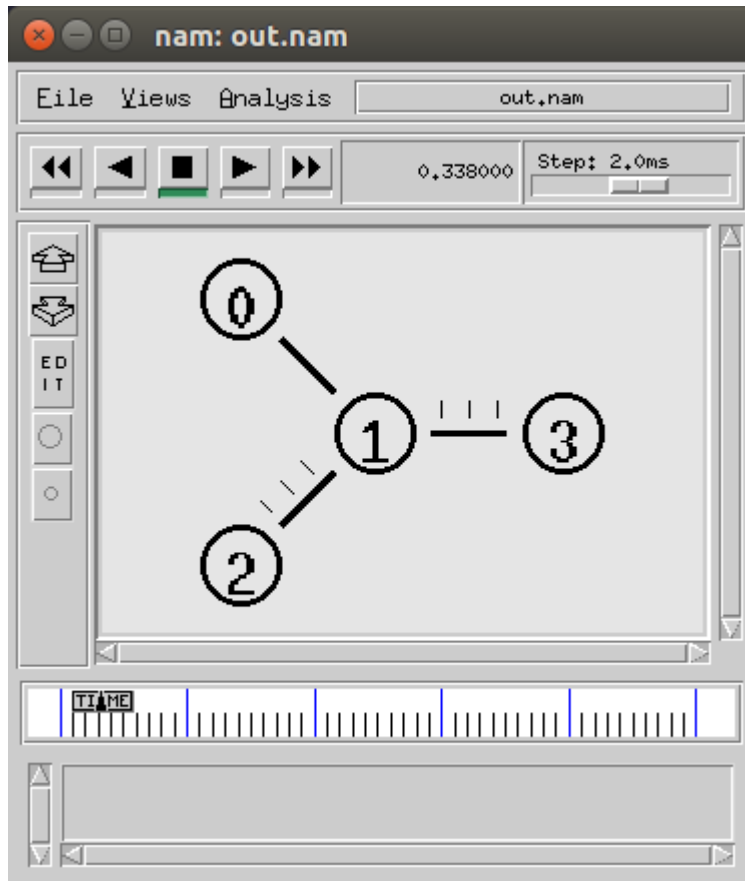
```
set udp [new Agent/UDP]
$ns attach-agent $n2 $udp
```

```
set cbr [new Application/Traffic/CBR]
$cbr attach-agent $udp
```

```
set null [new Agent/Null]
$ns attach-agent $n3 $null
```

```
$ns connect $tcp $sink
$ns connect $udp $null
$ns rtmodel-at 1.0 down $n1 $n3
$ns rtmodel-at 2.0 up $n1 $n3
$ns rtpeto LS
$ns at 0.0 "$ftp start"
$ns at 0.0 "$cbr start"
$ns at 5.0 "finish"
$ns run
```

Output:



Result:

Thus, the simulation program for Link State routing protocol is done using NS2

EXP: 9 PERFORMANCE EVALUATION OF ROUTING PROTOCOLS DATE: 30-09-2021

Aim:

To Study of performance evaluation of routing protocols using simulation tool

ROUTING PROTOCOLS

There are many routing protocols available. Among them all we are working with AODV and DSR for performance analysis.

A. Ad-hoc On demand Distance Vector (AODV)

It is purely On-Demand route acquisition routing protocol. It is better protocol than DSDV network as the size of network may increase depending on the number of vehicle nodes.

1) **Path Discovery Process:** In order to discover the path between source and destination, a Route Request message (RREQ) is broadcasted to all the neighbours in radio range who again continue to send the same to their neighbours in their radio range, until the destination is reached. Every node maintains two counters: sequence number and broadcast-id in order to maintain loop-free and most recent route information. The broadcast-id is incremented for every RREQ the source node initiates. If an intermediate node receives the same copy of request, it discards it without routing it further. When a node forwards the RREQ message, it records the address of the neighbour from which it received the first copy of the broadcast packet, in order to maintain a reverse path to the source node. The RREQ packet contains: the source sequence number and the last destination sequence number known to the source. The source sequence number is used to maintain information about reverse route and destination sequence number tells about the actual distance to the final node.

2) **Route Maintenance:** A source node sends a new moving request packet RREQ to find a new route to the destination. But, if an intermediate node moves from its place, its upstream neighbor noticed the move and sends a message notification failure of the link to each of its active upstream neighbors to inform them about the move to source nodes is achieved. After the detection process is again initiated.

B. Dynamic Source Routing (DSR)

It is an On-Demand routing protocol in which the sequence of nodes through which a packet needs to travel is calculated and maintained as an information in packet header. Every mobile node in the network needs to maintain a route cache where it caches source routes that it has learned. When a packet is sent, the route-cache inside the node is compared with the actual route needs to be covered.

1. **Route Discovery:** The source node broadcasts request-packets to all the neighbours in the network containing the address of the destination node, and a reply is sent back to the source node with the list of network-nodes through which it should propagate in the

process. Sender initiates the route record as a list with a single element containing itself followed by the linking of its neighbour in that route. A request packet also contains an identification number called request-id, which is counter increased only when a new route request packet is being sent by the source node. To make sure that no loops occur during broadcast, the request is processed in the given order. A route reply is obtained in DSR by two ways: Symmetric-links (bidirectional), in which the backward route is followed again to catch the source node. Asymmetric-links (unidirectional) needs to discover the route up to the source node in the same manner as the forward route is discovered.

2. **Route Maintenance:** In the hop by hop acknowledgement at data link layer allows the early detection and retransmission of lost or corrupt packets in the data-link layer. If a transmission error occurs, a route error packet containing the address of node detecting the error and the host address is sent back to the sender. Whenever a node receives a route error packet, the hop in error is removed from the route cache and all routes containing this hop are truncated at that point. When the wireless transmission between two nodes does not work equally well in both directions, and then end-to-end replies on the application or transport layer may be used to indicate the status of the route from one host to the other.

Result:

Thus, the performance evaluation of routing protocols was studied.

EXP: 10 SIMULATION OF ERROR CORRECTION CODE (LIKE CRC) DATE: 30-09-2021

Aim:

To write a Java program for Simulation of error correction code (like CRC)

Procedure:

1. Start the Program
2. Given a bit string, append 0s to the end of it (the number of 0s is the same as the degree of the generator polynomial) let $B(x)$ be the polynomial corresponding to B.
3. Divide $B(x)$ by some agreed on polynomial $G(x)$ (generator polynomial) and determine the remainder $R(x)$. This division is to be done using Modulo 2 Division.
4. Define $T(x) = B(x) - R(x)$
5. $(T(x)/G(x) \Rightarrow \text{remainder } 0)$
6. Transmit T, the bit string corresponding to $T(x)$.
7. Let T represent the bit stream the receiver gets and $T'(x)$ the associated polynomial. The receiver divides $T1(x)$ by $G(x)$. If there is a 0 remainder, the receiver concludes $T = T'$ and no error occurred otherwise, the receiver concludes an error occurred and requires a retransmission
8. Stop the Program

Program:

```
import java.io.*;
class crc_gen
{
    public static void main(String args[]) throws IOException
    {
        BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
        int[] data;
        int[] div;
        int[] divisor;
        int[] rem;
        int[] crc;
        int data_bits, divisor_bits, tot_length;

        System.out.println("Enter number of data bits : ");
        data_bits=Integer.parseInt(br.readLine());
        data=new int[data_bits];

        System.out.println("Enter data bits : ");
        for(int i=0; i<data_bits; i++)
            data[i]=Integer.parseInt(br.readLine());

        System.out.println("Enter number of bits in divisor : ");
        divisor_bits=Integer.parseInt(br.readLine());
        divisor=new int[divisor_bits];
```

```

System.out.println("Enter Divisor bits : ");
for(int i=0; i<divisor_bits; i++)
    divisor[i]=Integer.parseInt(br.readLine());

System.out.print("Data bits are : ");
for(int i=0; i< data_bits; i++)
    System.out.print(data[i]);
System.out.println();

System.out.print("divisor bits are : ");
for(int i=0; i< divisor_bits; i++)
    System.out.print(divisor[i]);
System.out.println();
    tot_length=data_bits+divisor_bits-1;

div=new int[tot_length];
rem=new int[tot_length];
crc=new int[tot_length];
/*----- CRC GENERATION-----*/
for(int i=0;i<data.length;i++)
    div[i]=data[i];

System.out.print("Dividend (after appending 0's) are : ");
for(int i=0; i< div.length; i++)
    System.out.print(div[i]);
System.out.println();

for(int j=0; j<div.length; j++){
    rem[j] = div[j];
}

rem=divide(div, divisor, rem);

for(int i=0;i<div.length;i++)    {
    crc[i]=(div[i]^rem[i]);
}

System.out.println();
System.out.println("CRC code : ");
for(int i=0;i<crc.length;i++)
    System.out.print(crc[i]);

/*-----ERROR DETECTION-----*/
System.out.println();
System.out.println("Enter CRC code of "+tot_length+" bits : ");
for(int i=0; i<crc.length; i++)
    crc[i]=Integer.parseInt(br.readLine());

```

```

        System.out.print("crc bits are : ");
for(int i=0; i< crc.length; i++)
    System.out.print(crc[i]);
System.out.println();
for(int j=0; j<crc.length; j++){
    rem[j] = crc[j];
}

rem=divide(crc, divisor, rem);

for(int i=0; i< rem.length; i++)
{
    if(rem[i]!=0)
    {
        System.out.println("Error");
        break;
    }
    if(i==rem.length-1)
        System.out.println("No Error");
}

System.out.println("THANK YOU.... :)");
}

static int[] divide(int div[],int divisor[], int rem[])
{
    int cur=0;
    while(true)
    {
        for(int i=0;i<divisor.length;i++)
            rem[cur+i]=(rem[cur+i]^divisor[i]);

        while(rem[cur]==0 && cur!=rem.length-1)
            cur++;

        if((rem.length-cur)<divisor.length)
            break;
    }
    return rem;
}
}

```

Output:

```

C:\ Command Prompt
Microsoft Windows [Version 10.0.19041.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ctadmin>cd Nithila

C:\Users\ctadmin\Nithila>javac crc_gen.java

C:\Users\ctadmin\Nithila>java crc_gen
Enter number of data bits :
7
Enter data bits :
1
0
1
1
0
0
1
Enter number of bits in divisor :
3
Enter Divisor bits :
1
0
1
Data bits are : 1011001
divisor bits are : 101
Dividend (after appending 0's) are : 101100100

CRC code :
101100111
Enter CRC code of 9 bits :
1
0
1
1
0
0
0
1
1
crc bits are : 101100101
Error
THANK YOU.... :)
```

Result:

Thus, the implementation of NS2 simulation using Link state routing protocol has been successfully executed and implemented.

CONTENT BEYOND SYLLABUS

IMPLEMENTATION OF REMOTE COMMAND EXECUTION (RCE)

Aim:

To implement Remote Command Execution (RCE).

Algorithm:

Client Side

1. Establish a connection between the Client and Server.

```
Socket client=new Socket("127.0.0.1",6555);
```

2. Create instances for input and output streams.

```
Print Stream ps=new Print Stream(client.getOutputStream());
```

3. `BufferedReader br=new BufferedReader(new InputStreamReader(System.in));`

4. Enter the command in Client Window.

Send the message to its output

```
str=br.readLine();
```

```
ps.println(str);
```

Server Side

1. Accept the connection request by the client.

```
ServerSocket server=new ServerSocket(6555);
```

```
Sockets=server.accept();
```

2. Get the IP address from its input stream.

```
BufferedReader br1=new BufferedReader(new InputStreamReader(s.getInputStream()));
```

```
ip=br1.readLine();
```

3. During runtime execute the process

```
Runtime r=Runtime.getRuntime();
```

```
Process p=r.exec(str);
```

Program:

Client Side

```
import java.io.*;
```

```

import java.net.*;
class clientRCE
{
public static void main(String args[]) throws IOException
{
try
{
String str;Socket client=new Socket("127.0.0.1",6555);
PrintStream ps=new PrintStream(client.getOutputStream());
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
System.out.println("\t\t\tCLIENT WINDOW\n\n\t\tEnter TheCommand:");
str=br.readLine();
ps.println(str);
}
catch(IOException e)
{
System.out.println("Error"+e); }
}
}

```

Server Side

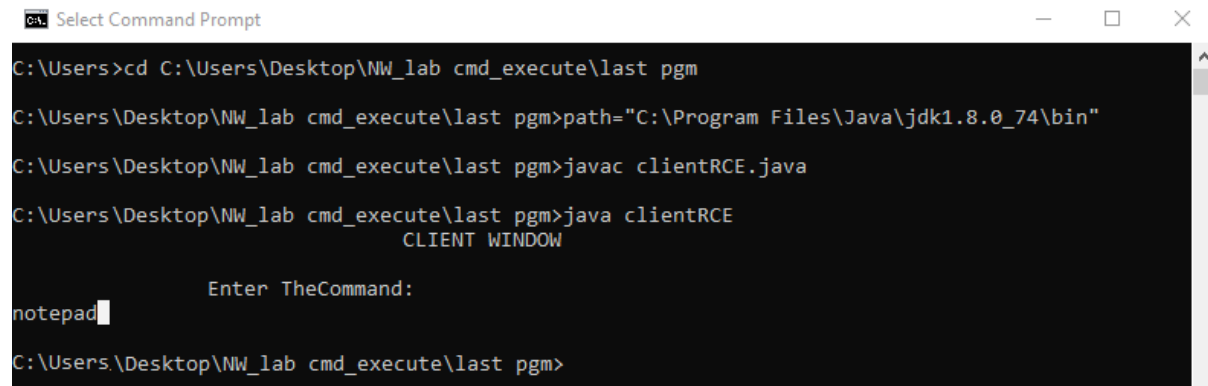
```

import java.io.*;
import java.net.*;
class serverRCE
{
public static void main(String args[]) throws IOException
{
try
{
String str;
ServerSocket server=new ServerSocket(6555);
Socket s=server.accept();
BufferedReader br=new BufferedReader(new InputStreamReader(s.getInputStream()));
str=br.readLine();
Runtime r=Runtime.getRuntime();
Process p=r.exec(str);
}
catch(IOException e)
{
System.out.println("Error"+e);
}
}
}

```

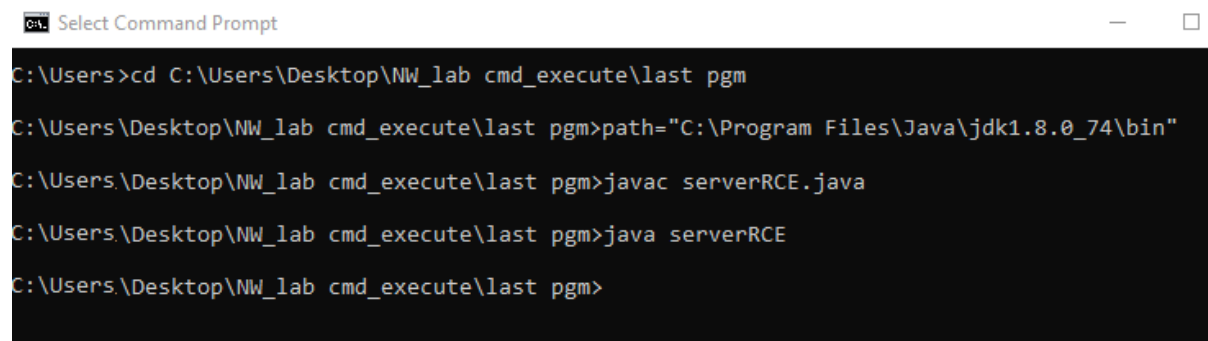

Output:

C:\NetworkingPrograms>java serverRCE



```
C:\Users>cd C:\Users\Desktop\NW_lab cmd_execute\last pgm
C:\Users\Desktop\NW_lab cmd_execute\last pgm>path="C:\Program Files\Java\jdk1.8.0_74\bin"
C:\Users\Desktop\NW_lab cmd_execute\last pgm>javac clientRCE.java
C:\Users\Desktop\NW_lab cmd_execute\last pgm>java clientRCE
CLIENT WINDOW
Enter TheCommand:
notepad
C:\Users\Desktop\NW_lab cmd_execute\last pgm>
```

C:\NetworkingPrograms>java clientRCE



```
C:\Users>cd C:\Users\Desktop\NW_lab cmd_execute\last pgm
C:\Users\Desktop\NW_lab cmd_execute\last pgm>path="C:\Program Files\Java\jdk1.8.0_74\bin"
C:\Users\Desktop\NW_lab cmd_execute\last pgm>javac serverRCE.java
C:\Users\Desktop\NW_lab cmd_execute\last pgm>java serverRCE
C:\Users\Desktop\NW_lab cmd_execute\last pgm>
```

Client Window

EnterTheCommand:

Notepad

Result:

Thus, the implementation RCE is done & executed successfully.