

Exercise 4: TCP

Aumrudh Lal Kumar TJ,18BIT034

BTech IT 3rd Year, 5th Sem

1)

Problem: Write a program for checking given string is palindrome or not using client, server in TCP.

Aim: To write a program for checking given string is palindrome or not using client, server in TCP in C and Java.

Program:

C

Server

```
#include<stdio.h>
#include<netdb.h>
#include<arpa/inet.h>
#include<netinet/in.h>
#include<stdlib.h>
#include<string.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<unistd.h>

int main(){
    int sd,b,l,ad,len;
    char msg[100],result[100];
```

```
struct sockaddr_in server,client;

server.sin_family=AF_INET;

printf("Enter the port no : ");

int portno;

scanf("%d",&portno);

server.sin_port=htons(portno);

server.sin_addr.s_addr=htonl(INADDR_ANY);
```

```
//socket creation
```

```
sd=socket(AF_INET,SOCK_STREAM,0);

if(sd==-1){

    printf("Socket creation failed\n");

    exit(0);

}

else{

    printf("Socket Created\n");

}
```

```
//binding
```

```
b=bind(sd,(struct sockaddr *)&server,sizeof(server));
```

```
if(b==-1){

    printf("Binding failed\n");

    exit(0);

}

else{

    printf("Binded\n");

}
```

```
}
```

```
//listen
```

```
l=listen(sd,5);
```

```
if(l==-1){
```

```
    printf("Listen failed\n");
```

```
    exit(0);
```

```
}
```

```
else{
```

```
    printf("Listening\n");
```

```
}
```

```
//accept
```

```
len=sizeof(client);
```

```
ad=accept(sd,(struct sockaddr*)&client,&len);
```

```
if(ad==-1){
```

```
    printf("Accept failed\n");
```

```
    exit(0);
```

```
}
```

```
else{
```

```
    printf("Accepted\n");
```

```
}
```

```
printf("Connection established\n");
```

```
do{
```

```
    recv(ad,msg,100,0);
```

```
    printf("Client side : %s\n",msg);
```

```
    int i=0;
```

```

    int j=strlen(msg)-1;
    while(j>i){
        if(msg[i++]!=msg[j--]){
            strcpy(result,msg);
            strcat(result," is not palindrome!");
            break;
        }
        strcpy(result,msg);
        strcat(result," is Palindrome!");
    }
    //scanf("%s",msg);
    send(ad,result,100,0);
}while(strcmp(msg,"bye")!=0);
close(sd);
}

```

Client

```

#include<stdio.h>
#include<netdb.h>
#include<arpa/inet.h>
#include<netinet/in.h>
#include<stdlib.h>
#include<string.h>
#include<sys/socket.h>
#include<sys/types.h>
#include<unistd.h>

```

```

int main(){
    int sd,c,len;
    char msg[100];
    struct sockaddr_in server,client;
    server.sin_family=AF_INET;
    printf("Enter the port no : ");
    int portno;
    scanf("%d",&portno);
    server.sin_port=htons(portno);
    server.sin_addr.s_addr=htonl(INADDR_ANY);

    //socket creation

    sd=socket(AF_INET,SOCK_STREAM,0);
    if(sd==-1){
        printf("Socket creation failed\n");
        exit(0);
    }
    else{
        printf("Socket Created\n");
    }

    //connection establishment

    len=sizeof(server);
    c=connect(sd,(const struct sockaddr*)&server,len);
    if(c==-1){
        printf("Connection failed\n");
        exit(0);
    }
}

```

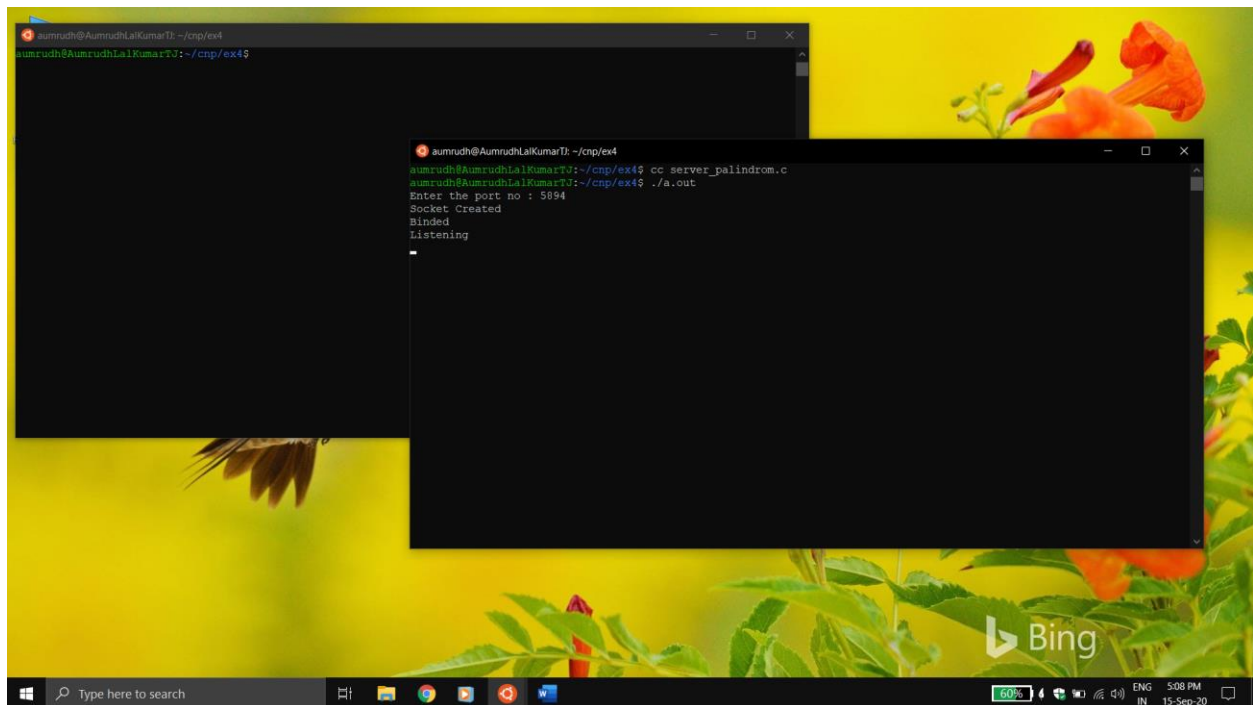
```

}
else{
    printf("Connection established\n");
}
do{
    printf("Enter client message : ");
    scanf("%s",msg);
    send(sd,msg,100,0);
    recv(sd,msg,100,0);
    printf("Server Message : %s\n",msg);
}while(strcmp(msg,"bye")!=0);
close(sd);
}

```

Output

Server Compilation



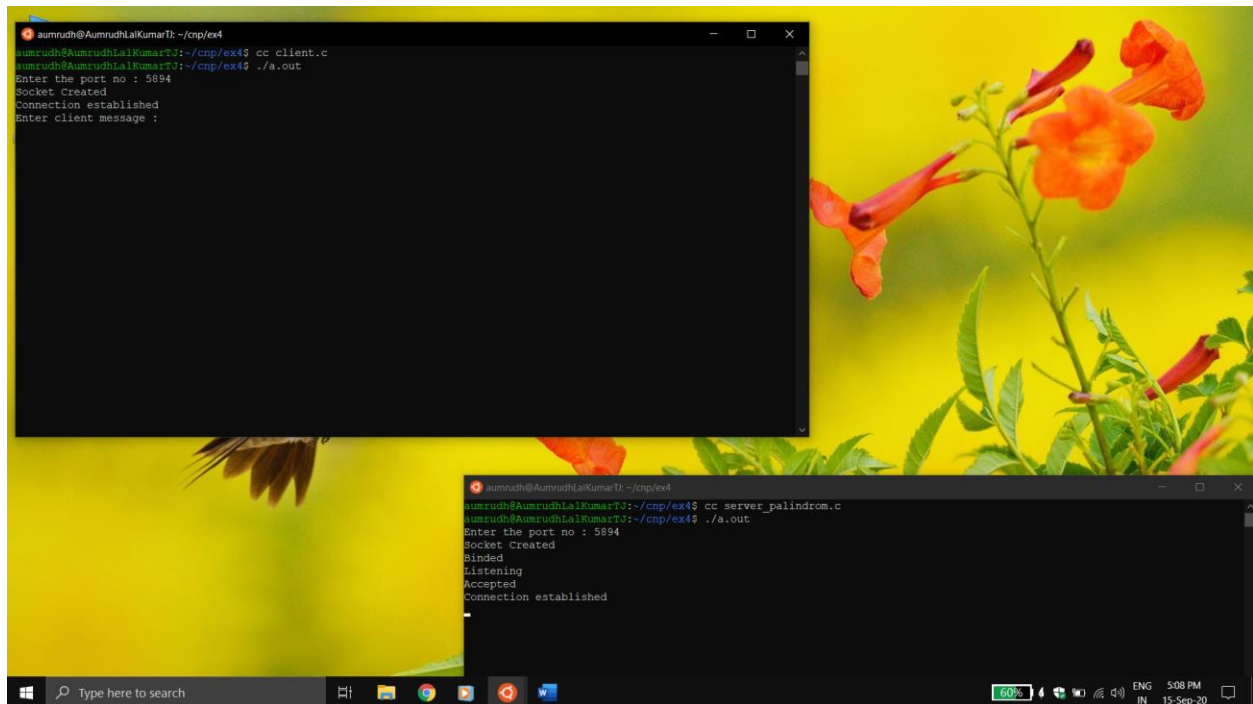
```

aumrudh@AumrudhLalKumarT2: ~/cnp/ex4
aumrudh@AumrudhLalKumarT2:~/cnp/ex4$

aumrudh@AumrudhLalKumarT2:~/cnp/ex4$ cc server_palindrom.c
aumrudh@AumrudhLalKumarT2:~/cnp/ex4$ ./a.out
Enter the port no : 5894
Socket Created
Binded
Listening

```

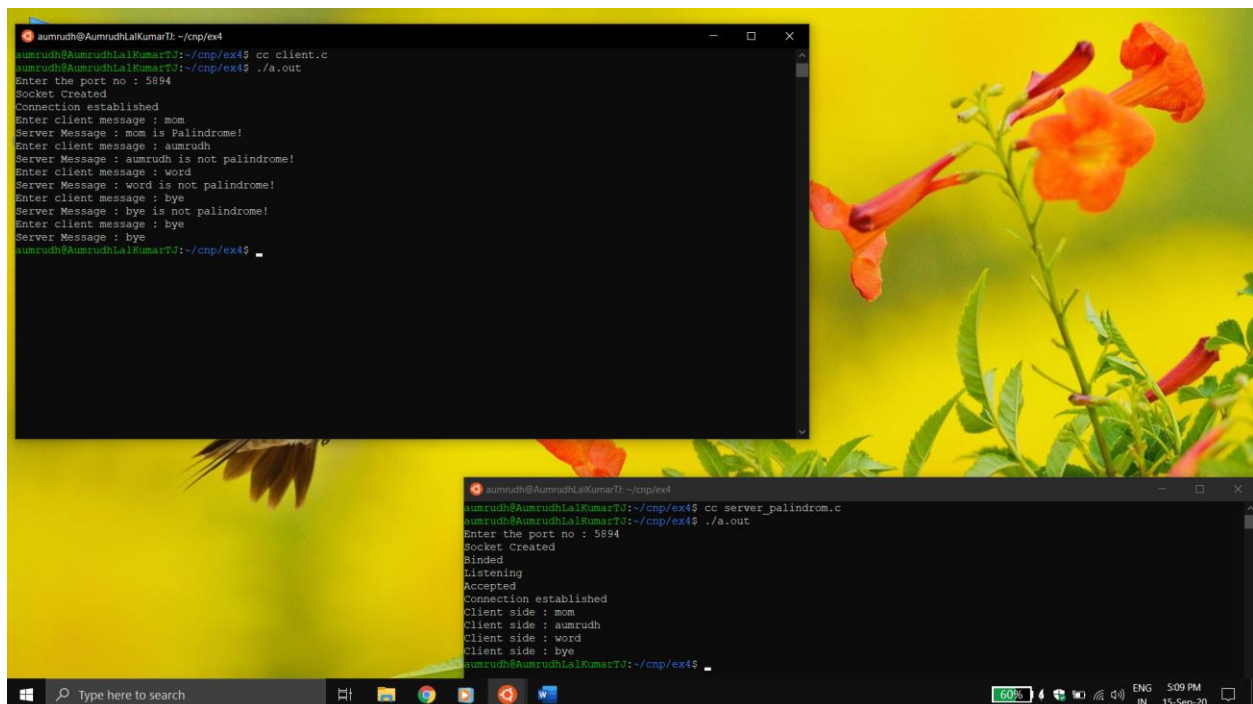
Client Compilation



```
aumrudh@AumrudhLalKumarTl: ~/cnp/ex4
aumrudh@aumrudhLalKumarTl:~/cnp/ex4$ cc client.c
aumrudh@aumrudhLalKumarTl:~/cnp/ex4$ ./a.out
Enter the port no : 5894
Socket Created
Connection established
Enter client message :
```

```
aumrudh@AumrudhLalKumarTl:~/cnp/ex4
aumrudh@aumrudhLalKumarTl:~/cnp/ex4$ cc server_palindrom.c
aumrudh@aumrudhLalKumarTl:~/cnp/ex4$ ./a.out
Enter the port no : 5894
Socket Created
Binded
Listening
Accepted
Connection established
```

Interactive message



```
aumrudh@AumrudhLalKumarTl: ~/cnp/ex4
aumrudh@aumrudhLalKumarTl:~/cnp/ex4$ cc client.c
aumrudh@aumrudhLalKumarTl:~/cnp/ex4$ ./a.out
Enter the port no : 5894
Socket Created
Connection established
Enter client message : mom
Server Message : mom is Palindrome!
Enter client message : aumrudh
Server Message : aumrudh is not palindrome!
Enter client message : word
Server Message : word is not palindrome!
Enter client message : bye
Server Message : bye is not palindrome!
Enter client message : bye
Server Message : bye
aumrudh@aumrudhLalKumarTl:~/cnp/ex4$
```

```
aumrudh@AumrudhLalKumarTl:~/cnp/ex4
aumrudh@aumrudhLalKumarTl:~/cnp/ex4$ cc server_palindrom.c
aumrudh@aumrudhLalKumarTl:~/cnp/ex4$ ./a.out
Enter the port no : 5894
Socket Created
Binded
Listening
Accepted
Connection established
Client side : mom
Client side : aumrudh
Client side : word
Client side : bye
aumrudh@aumrudhLalKumarTl:~/cnp/ex4$
```

Java

Server

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

public class Server{

    private int port;

    private Socket sd=null;

    private ServerSocket server= null;

    private DataInputStream in=null;


    public Server(int port){

        try{

            //creation of socket

            server=new ServerSocket(port);

            System.out.println("Server started");

            System.out.println("Waiting for a client ...");

            //accepting client request

            sd = server.accept();

            System.out.println("Client accepted");

            // takes input from the client socket

            in = new DataInputStream(new BufferedInputStream(sd.getInputStream()));

            String msg = "";

            // reads message from client until "bye" is sent
```



```

while (!msg.equals("bye")){
    try{
        msg = in.readUTF();
        System.out.println("Client Message : "+msg);
        int i = 0, j = msg.length() - 1;
        String result=msg;
        //check for palindrom
        while (i < j) {
            if (msg.charAt(i++) != msg.charAt(j--)){
                result=" is not palindrome!";
                break;
            }
            result=" is palindrome!";
        }
        //send result to client
        PrintWriter out=new PrintWriter(sd.getOutputStream(),true);
        out.println(result);
    }
    catch(Exception i){
        System.out.println(i);
    }
}

System.out.println("Closing connection");
sd.close();

    }

catch(Exception i){
    System.out.println(i);
}

```

```

    }

    public static void main(String args[]){
        Scanner input=new Scanner(System.in);
        System.out.print("Enter port no : ");
        int portno=input.nextInt();
        Server obj=new Server(portno);
    }
}

```

Client

```

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

public class Client{
    private static final String Server_IP="127.0.0.1";
    private int port;
    private Socket sd=null;
    private ServerSocket server=null;
    private DataInputStream input= null;
    private DataOutputStream out = null;
    Client(int port){
        try{
            sd=new Socket(Server_IP,port);
            System.out.println("Connected");
            // takes input from terminal
            input = new DataInputStream(System.in);

```

```

        // sends output to the socket
        out = new DataOutputStream(sd.getOutputStream());
    }
    catch(Exception i)
    {
        System.out.println(i);
    }

        // string to read message from input

String line = "";

// keep reading until "bye" is input
while (!line.equals("bye")){
    try{
        System.out.print("Enter the word : ");
        line = input.readLine();
        //send input data to server for processing
        out.writeUTF(line);
        //read input got from server
        BufferedReader ip=new BufferedReader(new
InputStreamReader(sd.getInputStream()));
        String serverResponse=ip.readLine();
        System.out.println("Server output : "+line+serverResponse);

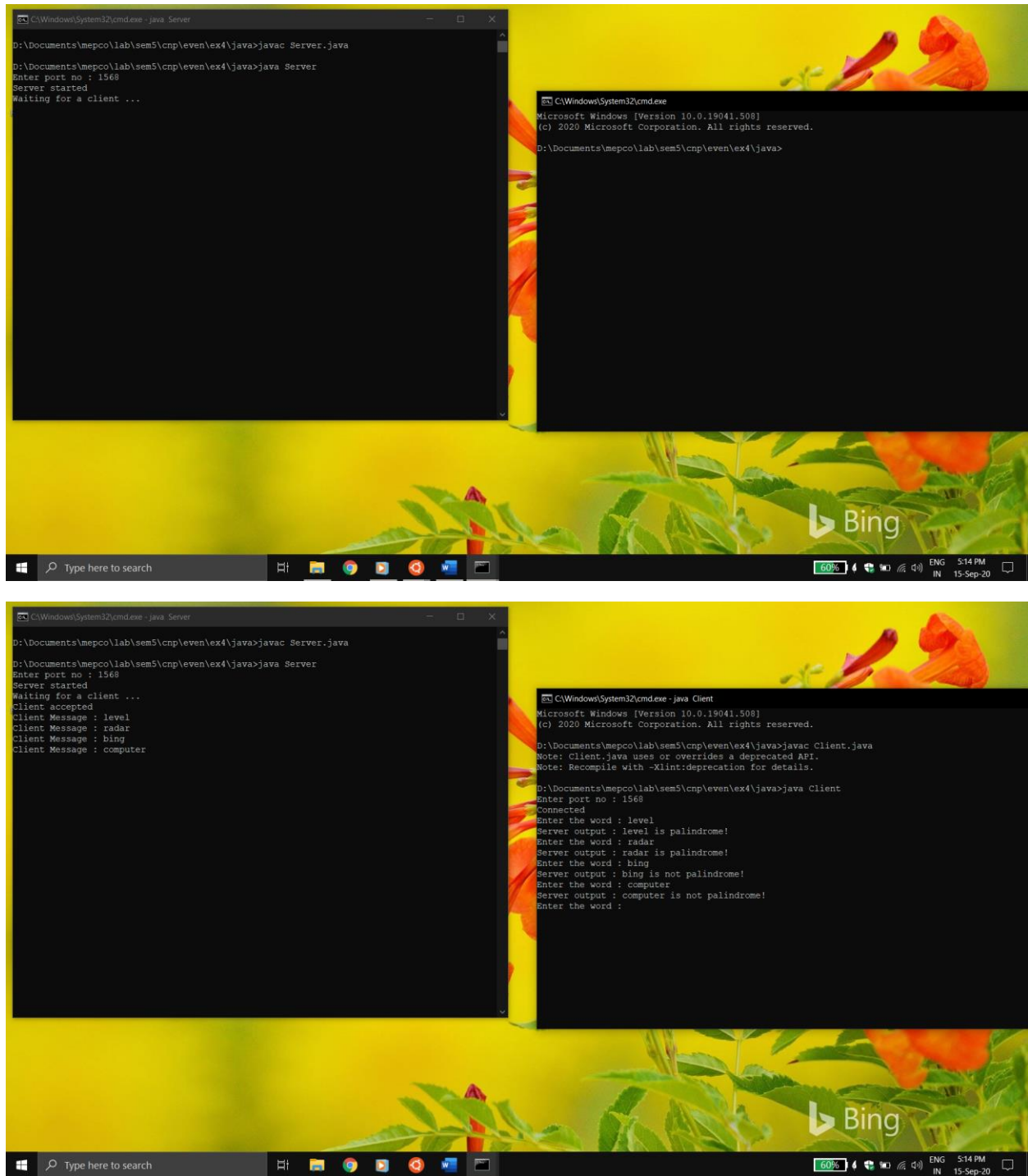
    }
    catch(IOException i){
        System.out.println(i);
    }
}

```

```
        try{
            input.close();
            out.close();
            sd.close();
        }
        catch(IOException i){
            System.out.println(i);
        }
    }

    public static void main(String args[]){
        Scanner input=new Scanner(System.in);
        System.out.print("Enter port no : ");
        int portno=input.nextInt();
        Client obj=new Client(portno);
    }
}
```

Output



The image displays two screenshots of a Windows desktop environment, showing the execution of a Java server and client program. The desktop background features a vibrant yellow and green floral pattern. The taskbar at the bottom includes the Windows Start button, a search bar, and several application icons. The system tray on the right shows the battery level at 60%, network status, and the date and time as 5:14 PM on 15-Sep-20.

Top Screenshot: Two command prompt windows are open. The left window, titled "C:\Windows\System32\cmd.exe - java Server", shows the compilation and execution of a server program. The right window, titled "C:\Windows\System32\cmd.exe", shows the execution of the client program.

```
C:\Windows\System32\cmd.exe - java Server
D:\Documents\mepco\lab\sem5\cnp\even\ex4\java>javac Server.java
D:\Documents\mepco\lab\sem5\cnp\even\ex4\java>java Server
Enter port no : 1568
Server started
Waiting for a client ...
```

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19041.508]
(c) 2020 Microsoft Corporation. All rights reserved.

D:\Documents\mepco\lab\sem5\cnp\even\ex4\java>
```

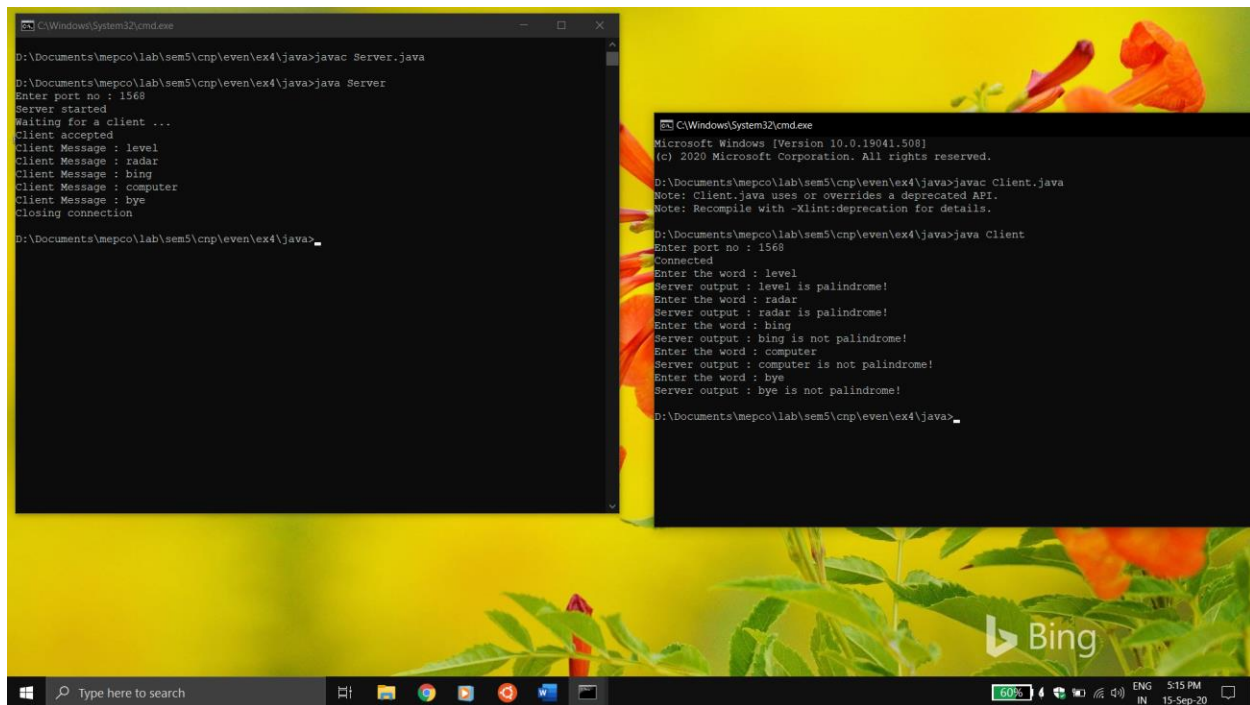
Bottom Screenshot: The same two command prompt windows are shown, but with more output. The server window now shows a client connection and three messages. The client window shows the execution of the client program, including port specification and word input/output.

```
C:\Windows\System32\cmd.exe - java Server
D:\Documents\mepco\lab\sem5\cnp\even\ex4\java>javac Server.java
D:\Documents\mepco\lab\sem5\cnp\even\ex4\java>java Server
Enter port no : 1568
Server started
Waiting for a client ...
Client accepted
Client Message : level
Client Message : radar
Client Message : bing
Client Message : computer
```

```
C:\Windows\System32\cmd.exe - java Client
Microsoft Windows [Version 10.0.19041.508]
(c) 2020 Microsoft Corporation. All rights reserved.

D:\Documents\mepco\lab\sem5\cnp\even\ex4\java>javac Client.java
Note: Client-java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.

D:\Documents\mepco\lab\sem5\cnp\even\ex4\java>java Client
Enter port no : 1568
Connected
Enter the word : level
Server output : level is palindrome!
Enter the word : radar
Server output : radar is palindrome!
Enter the word : bing
Server output : bing is not palindrome!
Enter the word : computer
Server output : computer is not palindrome!
Enter the word :
```



```
C:\Windows\System32\cmd.exe
D:\Documents\mepco\lab\sem5\cnp\even\ex4\java>javac Server.java
D:\Documents\mepco\lab\sem5\cnp\even\ex4\java>java Server
Enter port no : 1568
Server started
Waiting for a client ...
Client accepted
Client Message : level
Client Message : radar
Client Message : bingo
Client Message : computer
Client Message : bye
Closing connection
D:\Documents\mepco\lab\sem5\cnp\even\ex4\java>_

C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19041.508]
(c) 2020 Microsoft Corporation. All rights reserved.
D:\Documents\mepco\lab\sem5\cnp\even\ex4\java>javac Client.java
Note: Client.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
D:\Documents\mepco\lab\sem5\cnp\even\ex4\java>java Client
Enter port no : 1568
Connected
Enter the word : level
Server output : level is palindrome!
Enter the word : radar
Server output : radar is palindrome!
Enter the word : bingo
Server output : bingo is not palindrome!
Enter the word : computer
Server output : computer is not palindrome!
Enter the word : bye
Server output : bye is not palindrome!
D:\Documents\mepco\lab\sem5\cnp\even\ex4\java>_
```

Result:

The TCP/IP program for client, server for checking whether given string is palindrome or not is executed successfully using C and Java.