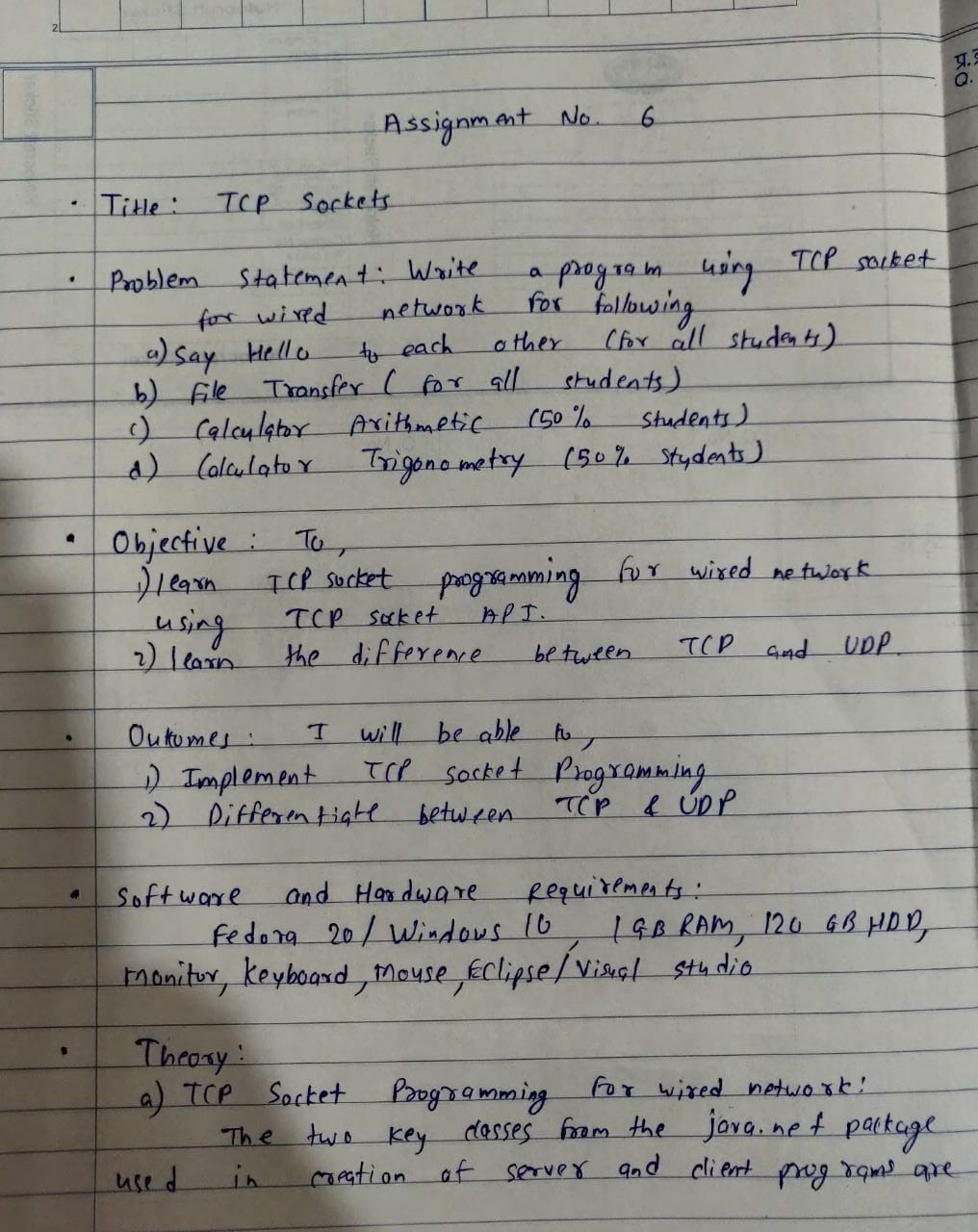
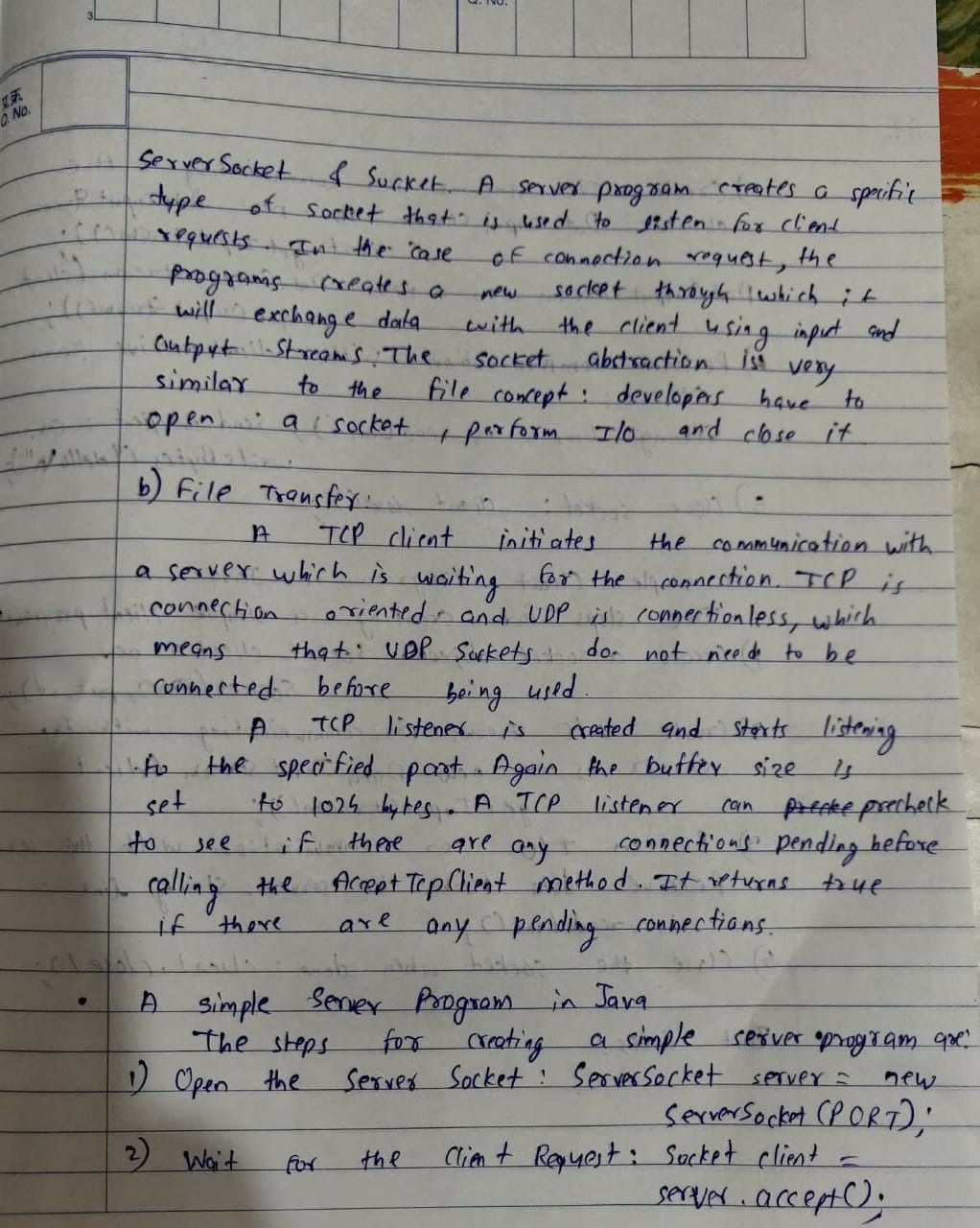
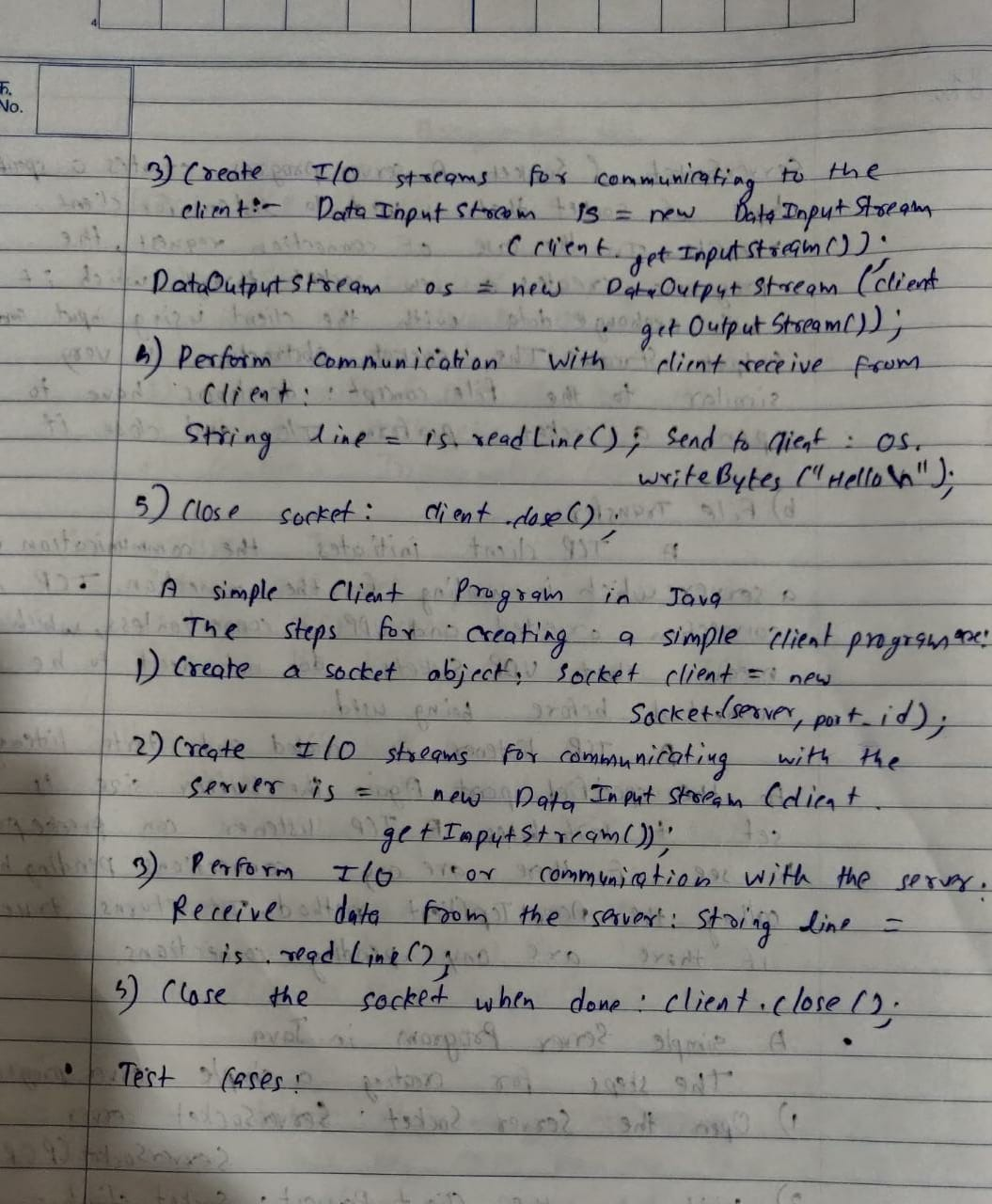
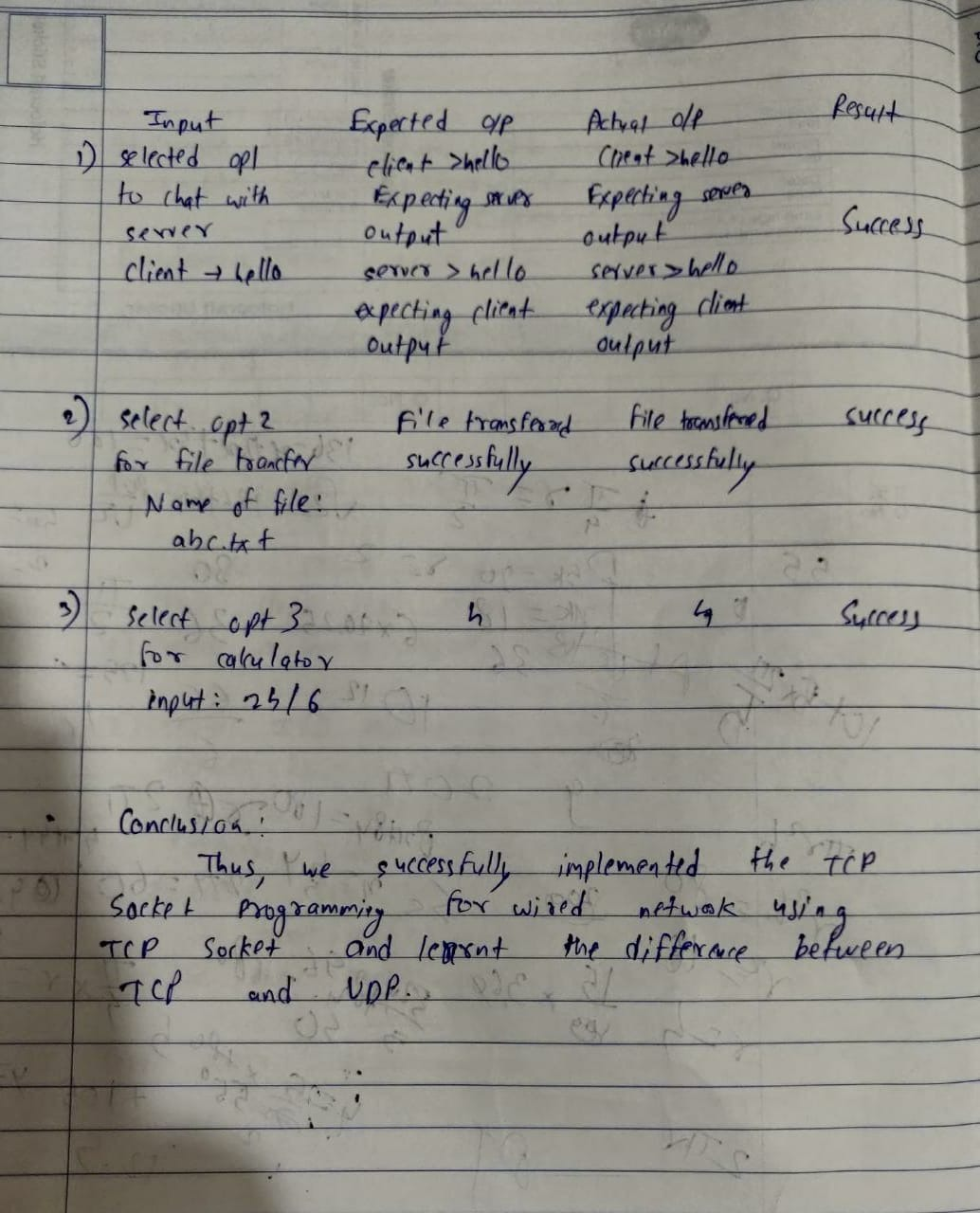
**CNL Assignment 6 (TCP Sockets) (31139-Durvesh)**

**// Write-ups**









**// Sample Code**

-------server.c

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <unistd.h>

#include <sys/types.h>

#include <sys/socket.h>

#include <netinet/in.h>

void error(const char \*msg) {

perror(msg);

exit(1);

}

int main(int argc, char const \*argv[]) {

if(argc<2){

fprintf(stderr,"Port number not provided.\n");

exit(1);

}

int sockfd, newsockfd, portno, n;

char buffer[255]; // The data to be sent to the server and received from it

struct sockaddr\_in serv\_addr, cli\_addr;

socklen\_t clilen;

sockfd = socket(AF\_INET, SOCK\_STREAM, 0); //SOCK\_STREAM for TCP

if(sockfd<0){

error("Error opening the server socket");

}

bzero((char \*)&serv\_addr,sizeof(serv\_addr));

portno = atoi(argv[1]);

serv\_addr.sin\_family = AF\_INET;

serv\_addr.sin\_addr.s\_addr = INADDR\_ANY;

serv\_addr.sin\_port = htons(portno); // Host to network short

if(bind(sockfd,(struct sockaddr \*) &serv\_addr,sizeof(serv\_addr))<0){

error("Binding Failed.");

}

listen(sockfd, 5); // Waiting for client to connect

clilen = sizeof(cli\_addr);

newsockfd = accept(sockfd,(struct sockaddr \*)&cli\_addr,&clilen); // Client accepted

if(newsockfd < 0)

error("Error accepting");

if(portno==9890){ // Chat App from server to client and vice versa

while(1){

bzero(buffer,255);

n = read(newsockfd,buffer,255); // Reading from client

if(n<0)

error("Error on reading");

printf("Client: %s\n",buffer);

bzero(buffer,255);

fgets(buffer,255,stdin); // Reading from the server terminal

n = write(newsockfd,buffer,strlen(buffer)); // Sending to CLient

if(n<0)

error("Error on writing\n");

int i = strncmp("bye",buffer,3);

if (i==0)

break;

}

}

if(portno==9891){ // File Transfer

FILE \*fp;

int ch =0;

fp= fopen("receivedtextfile.txt","a"); // Append if the file already exists or else create a new one

int words;

read(newsockfd,&words,sizeof(int)); // Read the word count

while (ch!=words) {

read(newsockfd,buffer,255); // Read a word

fprintf(fp,"%s ",buffer); // Write that word to the file

ch++;

}

printf("The file was received");

}

if(portno==9892){

int num1,num2,answer,choice;

char choices[5][15]={"Addition","Subtraction","Multiplication","Division","Exit"};

S: n = write(newsockfd,"Enter number 1: ",strlen("Enter number 1: ")); // Sending a message to ask for num1

if(n<0)

error("Error on writing\n");

read(newsockfd,&num1,sizeof(int)); // Reading num1 sent from client

printf("Client number 1 is: %d\n",num1);

n = write(newsockfd,"Enter number 2: ",strlen("Enter number 2: ")); // Sending a message to ask for num2

if(n<0)

error("Error on writing\n");

read(newsockfd,&num2,sizeof(int)); // Reading num2 sent from client

printf("Client number 2 is: %d\n",num2);

n = write(newsockfd,"1. Addition\n2. Subtraction\n3. Multiplication\n4. Division\n5. Exit\n",

strlen("1. Addition\n2. Subtraction\n3. Multiplication\n4. Division\n5. Exit\n"));//Sending request for choice

if(n<0)

error("Error on writing\n");

read(newsockfd,&choice,sizeof(int)); // Reading choice for operation

printf("Client operation is: %s\n",choices[choice-1]); // Fetching the opearation name from string array

switch (choice) {

case 1:

answer = num1+num2; //addition

break;

case 2:

answer = num1-num2; //Subtraction

break;

case 3:

answer = num1\*num2; //Multiplication

break;

case 4:

answer = num1/num2; //Division

break;

case 5:

goto Q; //Exit

break;

}

write(newsockfd,&answer,sizeof(int));

if(choice!=5){ //Exit Case

goto S;

}

}

Q: close(newsockfd);

close(sockfd);

return 0;

}

--------client.c

/\*

filename server\_ipaddress portno

\*/

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <unistd.h>

#include <sys/types.h>

#include <sys/socket.h>

#include <netinet/in.h>

#include <netdb.h>

#include <ctype.h>

void error(const char \*msg) {

perror(msg);

exit(1);

}

int main(int argc, char const \*argv[]) {

int sockfd,portno,n;

struct sockaddr\_in serv\_addr;

struct hostent \*server;

char buffer[255];

if(argc<3){

fprintf(stderr, "usage %s hostname port\nport 9890 for chat\nport 9891 for file transfer\nport 9892 for calculator\n",argv[0]);

exit(1);

}

portno = atoi(argv[2]); // String to integer

sockfd = socket(AF\_INET,SOCK\_STREAM,0);

if(sockfd < 0)

error("Error opening Socket");

server = gethostbyname(argv[1]);

if(server == NULL)

fprintf(stderr,"Error, no such host");

bzero((char \*)&serv\_addr,sizeof(serv\_addr));

serv\_addr.sin\_family = AF\_INET;

bcopy((char\*)server->h\_addr,(char \*)&serv\_addr.sin\_addr.s\_addr,server->h\_length);

serv\_addr.sin\_port = htons(portno); // host to network short

if(connect(sockfd,(struct sockaddr \*)&serv\_addr,sizeof(serv\_addr))<0){ // Connecting to Server

error("Connection failed");

}

if(portno==9890){ // Chat App

while(1){

bzero(buffer,255);

fgets(buffer,255,stdin); // Reading from client terminal

int j = strncmp("bye",buffer,3);

if (j==0)

break;

n = write(sockfd,buffer,strlen(buffer)); // Sending to server

if(n<0)

error("Error on writing");

bzero(buffer,255);

n = read(sockfd,buffer,255); // Reading from server

if(n<0)

error("Error on reading");

printf("Server says: %s\n",buffer); // Printing server's sent message

int i = strncmp("bye",buffer,3);

if (i==0)

break;

}

}

if(portno==9891){ // File Transfer

FILE \*f;

int words = 0;

char c;

f = fopen("textfile.txt","r"); // Opening an already existing file to send its data to server

while((c= getc(f))!= EOF){ // Counting the number of words

fscanf(f,"%s",buffer);

if(isspace(c)||c=='\t')

words++;

}

write(sockfd,&words,sizeof(int)); // Sending the number of words

rewind(f); // Setting the file pointer at initial word

char ch;

while(ch!=EOF){

fscanf(f,"%s",buffer);

write(sockfd,buffer,255); // Sending words one by one

ch = fgetc(f);

}

printf("The file has been sent.\n");

}

if(portno==9892){

while(1){

int num1,num2,choice,answer;

bzero(buffer,255);

n = read(sockfd,buffer,255); // Reading first request from server

if(n<0)

error("Error reading");

printf("Server- %s",buffer); // printing first request

scanf("%d", &num1); // getting num1 from client terminal

n = write(sockfd,&num1,sizeof(int)); // sending num1 to server

if (n<0) {

error("Error writing\n");

}

bzero(buffer,255);

n = read(sockfd,buffer,255); // Reading first request from server

if(n<0)

error("Error reading");

printf("Server- %s",buffer); // printing first request

scanf("%d", &num2); // getting num1 from client terminal

n = write(sockfd,&num2,sizeof(int)); // sending num2 to server

if (n<0) {

error("Error writing\n");

}

bzero(buffer,255);

n = read(sockfd,buffer,255);

if(n<0)

error("Error reading");

printf("Server- %s\n",buffer);

scanf("%d", &choice); // Reading operation choice from client

n = write(sockfd,&choice,sizeof(int));

if (n<0) {

error("Error writing\n");

}

if(choice==5){ // Exit case

goto E;

break;

}

printf("Answer: ");

n = read(sockfd,&answer,sizeof(int)); // Getting the answer.

printf("%d\n",answer ); // printing the answer

}

}

E: close(sockfd);

return 0;

}

**// Outputs**

