13/09/21 CN LAB 2 SACHIN RAGHUL T

OBSERVATION 2019103573

IMPLEMENTATION OF TCP APPLICATION WITH CLIENT-SERVER

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>
#include<ctype.h>
#include<time.h>

void error(const char *msg)
{
  perror(msg);
  exit(1);
}
int main(int argc, char *argv[])
{
  int sockfd, newsockfd, portno;
  socklen_t clilen;
  char buffer[1024];
  struct sockaddr_in serv_addr, cli_addr;
  int n;
  if (argc<2){
    fprintf(stderr, "ERROR, no port provided");
    exit(1);
  }
  sockfd = socket(AF_INET, SOCK_STREAM, 0);
  if (sockfd<0)
     error("ERROR opening socket");

  bzero((char*)&serv_addr, sizeof(serv_addr));
  portno = atoi(argv[1]);
  serv_addr.sin_family = AF_INET;
  serv_addr.sin_addr = INADDR_ANY;
  serv_addr.sin_port = htons(portno);
```

```
if (bind(sockfd, (struct sockaddr *)&serv_addr,
                    sizeof(serv_addr))<0)
  error("ERROR on binding");
listen(sockfd, 5);
clilen = sizeof(cli_addr);
newsockfd = accept(sockfd, (struct, sockaddr *)&cli_addr,
                   &clilen);
if (newsockfd < 0)
  error("ERROR on accept");
int num1, num2, ans, choice;
s: n = write(newsockfd, "Enter num1:", strlen("Enter num1:"));
if (n<0).
  error("ERROR writing to socket");
read(newsockfd, &num1, sizeof(int));
printf("client - Number 1 is: %d\n", num1);

n = write(newsockfd, "Enter num2:", strlen("Enter num2:"));
if(n<0)
  error("ERROR writing to socket");
read(newsockfd, &num2, sizeof(int));
printf("client - Number 2 is: %d\n", num2);

n = write(newsockfd, "Enter your choice: \n
1. Add \n 2. Sub \n 3. Mult \n 4. Div \n 5. Exit \n",
strlen("Enter your choice: \n
1. Add \n 2. Sub \n 3. Mult \n 4. Div \n 5. Exit \n"));

read(newsockfd, &choice, sizeof(int));
printf("client - choice is: %d\n", choice);

switch(choice)
{
  case 1;
    ans = num1 + num2;
    break;
```

```
    case 2:
      ans = num1 - num2;
      break;
    case 3:
      ans = num1 * num2;
      break;
    case 4:
      ans = num1 / num2;
      break;
    case 5:
    goto Q;
      break;
  }
  printf("Answer : %d\n", ans);
  write(newsockfd, &ans, sizeof(int));
  if(choice != 5)
    goto S;

  Q: close(newsockfd);
     close(sockfd);
     return 0;
}
```

client.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>
#include<netdb.h>
#include<type.h>
#include<time.h>

#define PORT 7787
#define MAXLINE 1024
```

```
void error(const char *msg)
{
  perror(msg);
  exit(0);
}
int main(int argc, char *argv[])
{
  int sockfd, portno, n;
  struct sockaddr_in servaddr;
  struct hostent *server;
  char buffer[1024];
  if(argc<3)
  {
    fprintf(stderr, "usage %s hostname port\n", argv[0]);
    exit(0);
  }
  portno = atoi(argv[2]);
  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\n");
    exit(0);
  }
  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);
  if(connect(sockfd, (struct sockaddr *)&serv_addr,
             sizeof(serv_addr))<0)
    error("ERROR connecting");
```

```
    int num1, num2, ans, choice;
s : bzero(buffer, 256);
    n = read(sockfd, buffer, 255);
    if (n<0)
      error("ERROR reading from socket");
    printf("server - %s\n", buffer);
    scanf("%d", &num1);
    write(sockfd, &num1, sizeof(int));

    bzero(buffer, 256);
    n = read(sockfd, buffer, 255);
    if(n<0)
      error("ERROR reading from socket");
    printf("server - %s\n", buffer);
    scanf("%d", &num2);
    write(sockfd, &num2, sizeof(int));

    bzero(buffer, 256);
    n = read(sockfd, buffer, 255);
    if (n<0)
      error("ERROR reading from socket");
    printf("server - %s", buffer);
    scanf("%d", &choice);
    write(sockfd, &choice, sizeof(int));
    if choice == 5)
      goto q;
    read(sockfd, &ans, sizeof(int));
    printf("server - The answer is : %d\n", ans);
    if(choice != 5)
      goto s;
q : printf("You chose to exit. Thank You!");
    close(sockfd);
    return 0;
}
```

SAMPLE OUTPUT :

```
$gcc -o server server.c
./server 8400
client - Number 1 is : 2
client - Number 2 is : 3
client - choice is : 1
Answer : 5
client - Number 1 is : 0
client - Number 2 is : 0
client - choice is : 5
```

```
$gcc -o client client.c
./client 127.0.0.1 8400
Server - Enter number 1
2
Server - Enter number 2
3
Server - Enter your choice :
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Exit
1
Server - Enter Number 1
0
Server - Enter Number 2
0
Server - Enter your choice :
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Exit.
5
```