

END SEMESTER
**COMPUTER
NETWORKS LAB**

COE19B005 KADAMBI NARASIMHAN AKASH

27th Nov, 2021

Server Code

```
#include<stdio.h>

#include<stdlib.h>

#include<unistd.h>

#include<sys/types.h>

#include<sys/socket.h>

#include<netinet/in.h>

#include<string.h>

#define MAX 80

struct Train
{
    int Trnum;

    char Trname[MAX];

    char From[MAX];

    char To[MAX];

}Trains[MAX];

struct Trainav
{
    int Trnum;

    int available;
```

```
}Trainsav[MAX];

int Trsize;

void message(int cid)
{
    int req,Trnum,Travailable;

    char msg[MAX];

    //Starts when Requested.

    read(cid,&req,sizeof(req));

    //Sends Size of Database

    write(cid,&Trsize,sizeof(Trsize));

    //Sends the database

    write(cid,Trains,Trsize*sizeof(struct Train));

    readagain:

    //Booking happens here one at a time

    //gets train number

    read(cid,&Trnum,sizeof(Trnum));
```

```
int flag;

flag=0;

int hold;

for(size_t i=0;i<Trsize;++i)
{
    if(Trainsav[i].Trnum==Trnum)
    {
        flag=1;

        hold=i;

        Travailable=Trainsav[i].available;

        break;
    }
}

if(flag==0)

    Travailable=-1;

write(cid,&Travailable,sizeof(Travailable));

if(flag==0)

    goto readagain;
```

```

    if(Travailable>0)
    {
        Trainsav[hold].available-=1;

        strcpy(msg,"Reservation Succesful");

        write(cid,msg,MAX);

    }

    else

    {

        strcpy(msg,"No Seats Available");

        write(cid,msg,MAX);

    }

    return ;
}

int main()
{

    int sid;

    if((sid=socket(AF_INET,SOCK_STREAM,0))== -1)

    {

        printf("Unable to Create a Server Socket\n");

        exit(1);
    }
}

```

```

}

struct sockaddr_in server;

memset(&server,0,sizeof(server));

server.sin_family=AF_INET;

server.sin_port=htons(1114);

server.sin_addr.s_addr=INADDR_ANY;


if(bind(sid,(struct sockaddr*)&server,sizeof(server))== -1)
{
    printf("Unable to Bind Server Socket\n");

    exit(1);
}


if(listen(sid,5)== -1)
{
    printf("Unable to Turn Server to passive mode\n");

    exit(1);
}


int cid;

struct sockaddr_in client;

int csize=sizeof(client);

```

```

cid=accept(sid, (struct sockaddr*)&client, &csize);

printf("Enter no.of Trains for Database\n");

scanf("%d", &Trsize);

for(size_t i=0; i<Trsize; ++i)
{
    printf("Enter the Details of Train %ld (Train num, Train Name, From
Station Name, To Station Name, Availabe Seats) :\n", i+1);

    scanf("%d %s %s
%s", &Trains[i].Trnum, Trains[i].Trname, Trains[i].From, Trains[i].To);

    scanf("%d", &Trainsav[i].available);

    Trainsav[i].Trnum=Trains[i].Trnum;
}

message(cid);

close(sid);

return 0;
}

```

Client Code

```
#include<stdio.h>

#include<stdlib.h>

#include<unistd.h>

#include<sys/types.h>

#include<sys/socket.h>

#include<netinet/in.h>

#include<string.h>

#define MAX 80

struct Train
{
    int Trnum;

    char Trname[MAX];

    char From[MAX];

    char To[MAX];

}Trains[MAX];

void displaytrains(struct Train* Trains,int Trsize)
{
    printf("-----Train
Details-----\n");
```



```

    for(size_t i=0;i<Trsize;++i)

    {

        printf("%d %s %s
%s\n",Trains[i].Trnum,Trains[i].Trname,Trains[i].From,Trains[i].To);

    }

printf("-----\n");

    return ;

}

void message(int sid)

{

    int req,Trsize,Trnum,Travailable;

    char msg[MAX];

    printf("Enter 1 to send database request\nEnter 0 to stop\n");

    scanf("%d",&req);

    if(req==0)

        return ;

    //Puts request

    write(sid,&req,sizeof(req));

```

```
//Reads size of Database

read(sid,&Trsize,sizeof(Trsize));

//Reads Database

read(sid,Trains,Trsize*sizeof(struct Train));

displaytrains(Trains,Trsize);

writeagain:

//Booking happens here one at a time

printf("Enter the Train number to book a seat for\n");

scanf("%d",&Trnum);

//writes the Train to book

write(sid,&Trnum,sizeof(Trnum));

//reads Train number validity

read(sid,&Travailable,sizeof(Travailable));

if(Travailable<0)

{

    printf("Invalid train Number, Please Reenter\n");
```

```

        goto writeagain;

    }

    //Booking Status read

    read(sid,msg,MAX);

    printf("Recieved From Server: %s\n",msg);

    return ;
}

int main()
{

    int sid;

    if((sid=socket(AF_INET,SOCK_STREAM,0))==-1)

    {

        printf("Unable to get the Server Socket\n");

        exit(1);

    }

    struct sockaddr_in server;

    memset(&server,0,sizeof(server));

    server.sin_family=AF_INET;

    server.sin_port=htons(1114);

```

```
server.sin_addr.s_addr=INADDR_ANY;

if(connect(sid, (struct sockaddr*)&server, sizeof(server))==-1)

{

    printf("Unable to connect to the Server\n");

    exit(0);

}

message(sid);

close(sid);

return 0;

}
```

Outputs

```
akashkadambi@akashkadambi: /mnt/c/Users/Adithya/Desktop/SEM_5_CSE/CNP/LAB3
akashkadambi@akashkadambi: /mnt/c/Users/Adithya/Desktop/SEM_5_CSE/CNP/LAB3$ gcc -o s servertcp.c
Adithya@akashkadambi: /mnt/c/Users/Adithya/Desktop/SEM_5_CSE/CNP/LAB3$ ./s
Enter no. of Trains for Database
3
Enter the Details of Train 1 (Train num, Train Name, From Station Name, To Station Name, Available Seats) :
1 a x y 5
Enter the Details of Train 2 (Train num, Train Name, From Station Name, To Station Name, Available Seats) :
2 b w z 1
DemoEnter the Details of Train 3 (Train num, Train Name, From Station Name, To Station Name, Available Seats) :
3 c p q 0
akashkadambi@akashkadambi: /mnt/c/Users/Adithya/Desktop/SEM_5_CSE/CNP/LAB3$

akashkadambi@akashkadambi: /mnt/c/Users/Adithya/Desktop/SEM_5_CSE/CNP/LAB3
akashkadambi@akashkadambi: /mnt/c/Users/Adithya/Desktop/SEM_5_CSE/CNP/LAB3$ gcc -o c clienttcp.c
akashkadambi@akashkadambi: /mnt/c/Users/Adithya/Desktop/SEM_5_CSE/CNP/LAB3$ ./c
Enter 1 to send database request
Enter 0 to stop
1
-----Train Details-----
1 a x y
2 b w z
3 c p q
-----
Enter the Train number to book a seat for
5
Invalid train Number, Please Reenter
Enter the Train number to book a seat for
3
Recieved From Server: No Seats Available
akashkadambi@akashkadambi: /mnt/c/Users/Adithya/Desktop/SEM_5_CSE/CNP/LAB3$
```

```
akashkadambi@akashkadambi: /mnt/c/Users/Adithya/Desktop/SEM_5_CSE/CNP/LAB3
akashkadambi@akashkadambi: /mnt/c/Users/Adithya/Desktop/SEM_5_CSE/CNP/LAB3$ gcc -o s servertcp.c
Adithya@akashkadambi: /mnt/c/Users/Adithya/Desktop/SEM_5_CSE/CNP/LAB3$ ./s
Enter no. of Trains for Database
3
Enter the Details of Train 1 (Train num, Train Name, From Station Name, To Station Name, Available Seats) :
1 a x y 5
Enter the Details of Train 2 (Train num, Train Name, From Station Name, To Station Name, Available Seats) :
2 b w z 1
DemoEnter the Details of Train 3 (Train num, Train Name, From Station Name, To Station Name, Available Seats) :
3 c p q 0
akashkadambi@akashkadambi: /mnt/c/Users/Adithya/Desktop/SEM_5_CSE/CNP/LAB3$

akashkadambi@akashkadambi: /mnt/c/Users/Adithya/Desktop/SEM_5_CSE/CNP/LAB3
akashkadambi@akashkadambi: /mnt/c/Users/Adithya/Desktop/SEM_5_CSE/CNP/LAB3$ gcc -o c clienttcp.c
akashkadambi@akashkadambi: /mnt/c/Users/Adithya/Desktop/SEM_5_CSE/CNP/LAB3$ ./c
Enter 1 to send database request
Enter 0 to stop
1
-----Train Details-----
1 a x y
2 b w z
3 c p q
-----
Enter the Train number to book a seat for
1
Recieved From Server: Reservation Successful
akashkadambi@akashkadambi: /mnt/c/Users/Adithya/Desktop/SEM_5_CSE/CNP/LAB3$
```