**08/11/2021 CN LAB 10 2019103573**

**IMPLEMENTING THE WINDOW SLIDING PROTOCOL**

**SERVER**

#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 <arpa/inet.h>

#define PORT 4042

int main()

{

    int socketfd = 0, clientfd = 0;

    struct sockaddr\_in host\_addr, client\_addr;

    socklen\_t length = sizeof(struct sockaddr\_in);

    char buffer[1024];

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

    if (socketfd < 0)

    {

        fprintf(stderr, "Socket creation error\n");

        return -1;

    }

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

    host\_addr.sin\_port = ntohs(PORT);

    inet\_pton(AF\_INET, "127.0.0.1", &host\_addr.sin\_addr);

    if (bind(socketfd, (struct sockaddr \*)&host\_addr, length) < 0)

    {

        fprintf(stderr, "Error in bind creation\n");

        return -1;

    }

    if (listen(socketfd, 5) < 0)

    {

        fprintf(stderr, "Error in listen port\n");

        return -1;

    }

    fprintf(stdout,"Listening on %s : %d\n",inet\_ntoa(host\_addr.sin\_addr),ntohs(host\_addr.sin\_port));

    while(1){

        fprintf(stdout, "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n\n");

        clientfd = accept(socketfd, (struct sockaddr \*)&host\_addr, &length);

        if (clientfd < 0)

        {

            fprintf(stderr, "Error in accepting connection.\n");

            continue;

        }

        fprintf(stdout, "Accepted connection.\n");

        int w, f;

        recv(clientfd, &w, sizeof(int), 0);

        fprintf(stdout, "Window size : %d\n", w);

        recv(clientfd, &f, sizeof(int), 0);

        fprintf(stdout, "Number of frames to transmit : %d\n", f);

        int i = 0, frames;

    S:

        recv(clientfd, &frames, sizeof(int), 0);

        if (frames == -9999)

        {

            fprintf(stdout, "\nSending acknowledgement of above frames \n\n");

            fprintf(stdout,"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n\n");

            close(clientfd);

            break;

        }

        i++;

        fprintf(stdout, " %d\t", frames);

        if (i == w)

        {

            i = 0;

            fprintf(stdout, "\nSending acknowledgement of above frames\n\n");

        }

        goto S;

    }

    close(socketfd);

    return 0;

}

**CLIENT**

#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 <arpa/inet.h>

#define PORT 4042

int main()

{

    int socketfd = 0;

    struct sockaddr\_in host\_addr;

    socklen\_t length = sizeof(struct sockaddr\_in);

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

    if (socketfd < 0)

    {

        fprintf(stderr, "Error in creating socket.\n");

        return -1;

    }

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

    host\_addr.sin\_port = htons(PORT);

    inet\_pton(AF\_INET, "127.0.0.1", &host\_addr.sin\_addr);

    if (connect(socketfd, (struct sockaddr \*)&host\_addr, length) < 0)

    {

        fprintf(stderr, "Error in connecting to server.\n");

        return -1;

    }

fprintf(stdout, "Connection established.\n");

    int w, f;

    fprintf(stdout, "\nEnter Window size : ");

    scanf("%d", &w);

    send(socketfd, &w, sizeof(int), 0);

    fprintf(stdout, "\nEnter number of frames to transmit: ");

    scanf("%d", &f);

    send(socketfd, &f, sizeof(int), 0);

    int frames[f], temp[w];

    fprintf(stdout, "\nEnter %d frames : ", f);

    for (int i = 0; i < f; i++)

    {

        scanf("%d", &frames[i]);

    }

    int i, j, t;

    for (i = 0; i < f; i++)

    {

        if ((i + 1) % w == 0)

        {

            t = frames[i];

            send(socketfd, &t, sizeof(int), 0);

            fprintf(stdout, "%d\t", t);

            fprintf(stdout,"\nAcknowledgement of above frames sent is received by sender\n\n");

            j = 0;

        }

        else

        {

            t = frames[i];

            send(socketfd, &t, sizeof(int), 0);

            fprintf(stdout, "%d\t", t);

        }

    }

    if (f % w != 0)

        fprintf(stdout,"\nAcknowledgement of above frames sent is received by sender\n\n");

    t = -9999;

    send(socketfd, &t, sizeof(int), 0);

    close(socketfd);

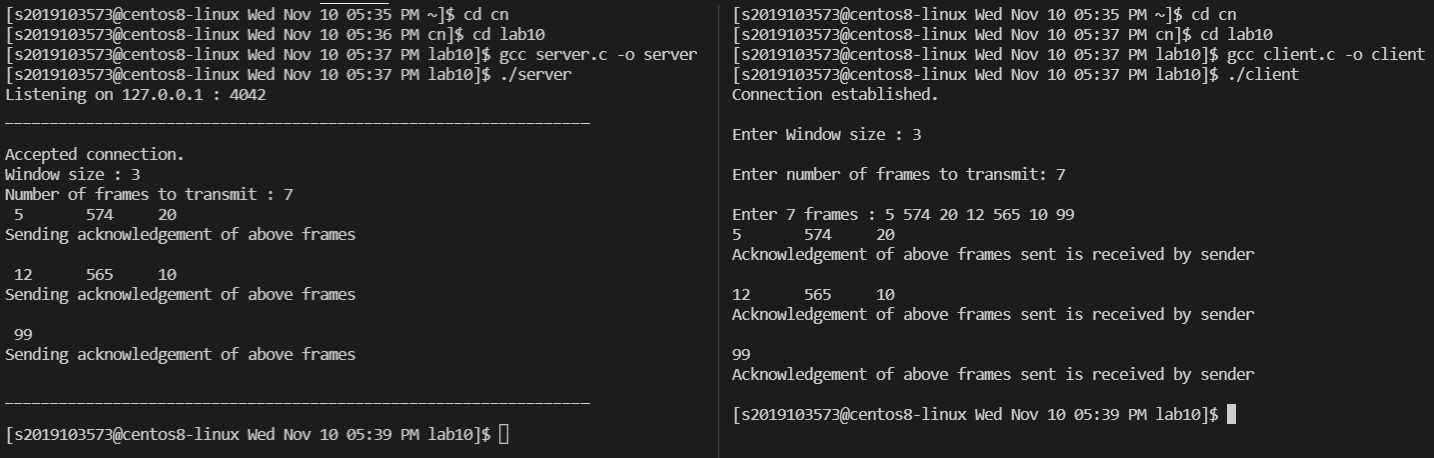
    return 0;

}

**OUTPUT :-**

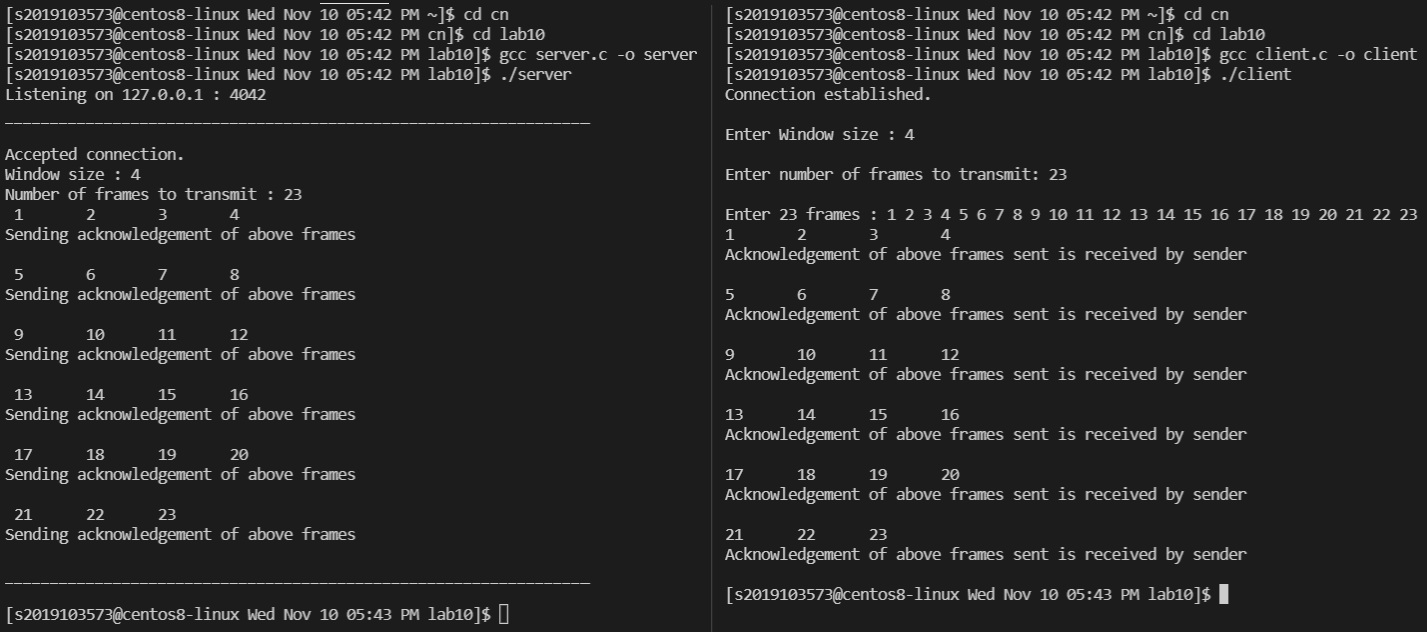
**1)**

**Window size : 3**

**No. Of Frames : 7**

**2)**

**Window size : 4**

**No. Of Frames : 23**