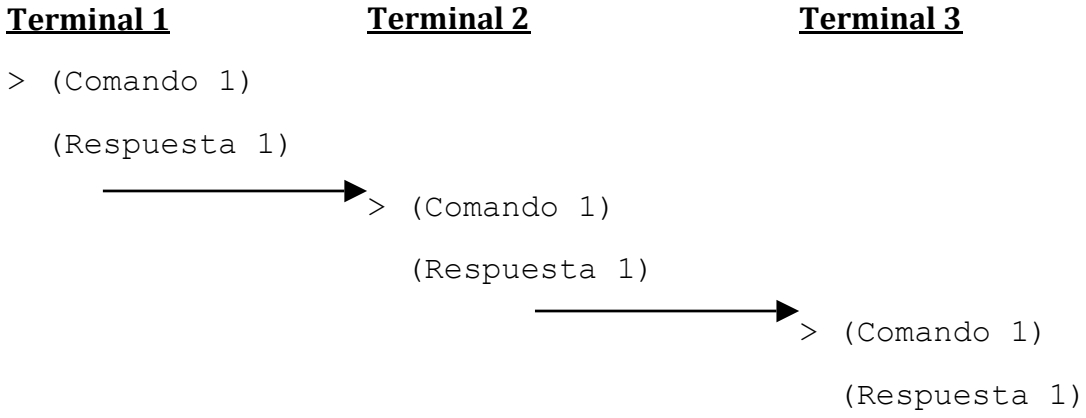
NOMBRE Y APELLIDOS DEL ALUMNO:

Arquitectura de Redes

Laboratorio

Prueba Final

1.- (0.75 puntos) Indicar la secuencia de comandos que permite descargar, por medio del protocolo FTP sobre la utilidad netcat, el archivo: /public/Docs/listado.txt en modo pasivo del servidor ftp.servidor.com. Si es preciso utilizar varios terminales, representar los comandos introducidos en cada uno y su precedencia de la siguiente manera.



Lista de comandos del protocolo FTP que pueden ser de utilidad.

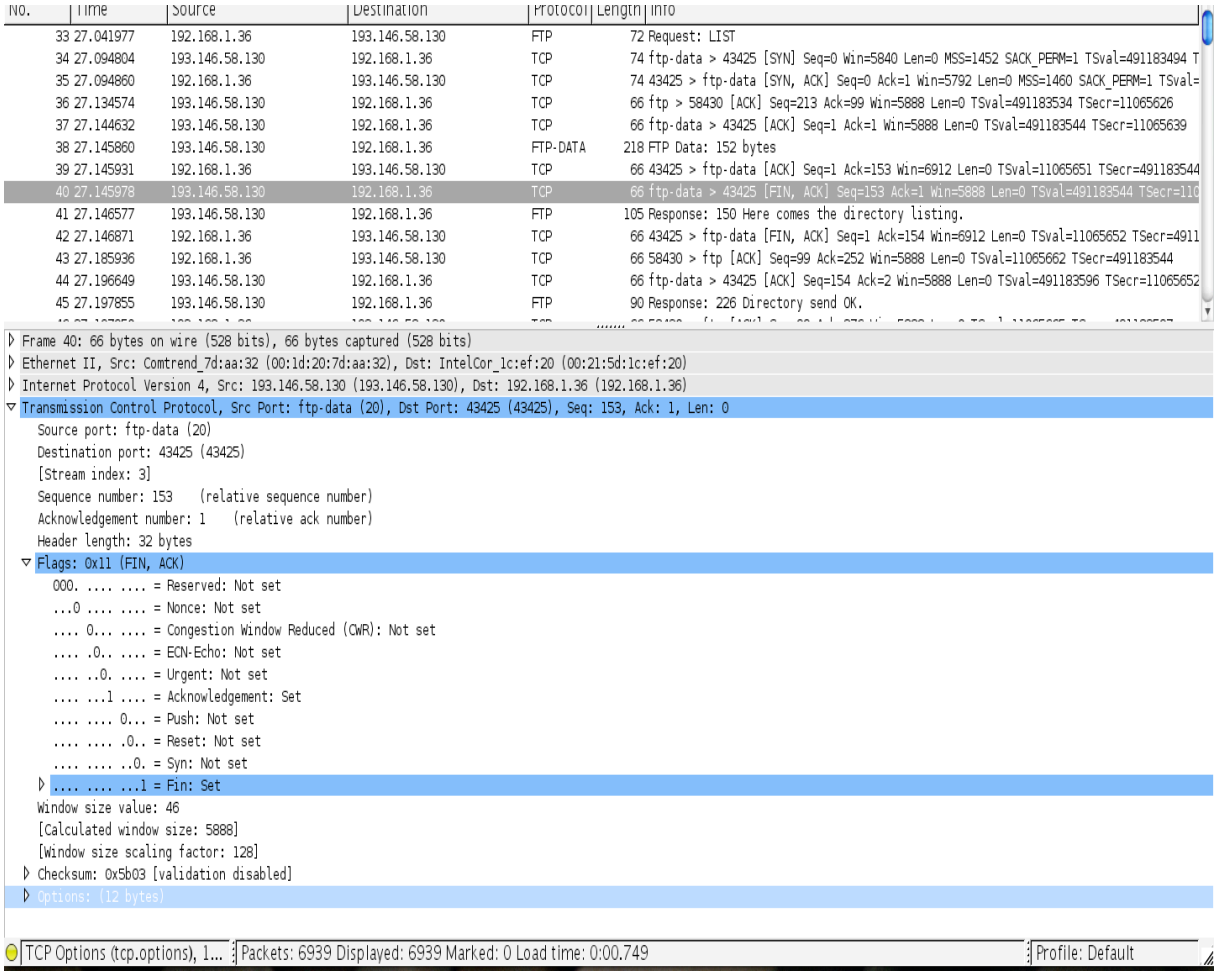
CWD		PASV	
EPRT	|1|a1,a2,a3,a4|puerto|	PORT	a1,a2,a3,a4,p1,p2
EPSV		RETR	
LIST		STAT	
PASS		USER	

2.- (0.75 puntos) El código siguiente es similar al del ejercicio 4 de la práctica 2. Se trata de un sencillo servidor que devuelve la suma de dos datos que el cliente envía. Se incluye el código del cliente y el del servidor. Modificar lo necesario (indicando qué y dónde) para que, en lugar de la suma, el servidor devuelva un 1 si es mayor el primero de los dos, un -1 si es mayor el segundo, y un 0 si son iguales. El cliente los mostrará por pantalla de mayor a menor según le indique el servidor.

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

int main(int argc, char *argv[])
{
  struct sockaddr_in server_addr,  client_addr;
  int c,i,sd, sc, val;
  unsigned int size;
  long int num[2], res;

  sd =  socket(PF_INET, SOCK_STREAM, IPPROTO_TCP);
  val = 1;
  setsockopt(sd, SOL_SOCKET, SO_REUSEADDR, (char *) &val, sizeof(int));

  bzero((char *)&server_addr, sizeof(server_addr));
  server_addr.sin_family = AF_INET;
  server_addr.sin_addr.s_addr = INADDR_ANY;
  server_addr.sin_port = htons(56789);

  bind(sd, (struct sockaddr *)&server_addr, sizeof(server_addr));
  listen(sd, 5);
  size = sizeof(client_addr);

  while (1)
    {
      printf("Esperando conexion\n");
      sc = accept(sd, (struct sockaddr *)&client_addr,&size);

      for(i=0;
          i<2*sizeof(long int)&&(c=read(sc,((char *)num)+i,2*sizeof(long int)-i))>0;
          i+=c);   /* recibe la petición */

    res = htonl(ntohl(num[0]) + ntohl(num[1]));
     write(sc, &res, sizeof(long int));     /*/ se envía el resultado     */
               close(sc);
    }
  close (sd);
  return 0;
}


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

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

```
  int i,c,sd;
  struct sockaddr_in server_addr;
  struct hostent *hp;
  long int num[2], res;

  if (argc!=3)
    {
     fprintf(stderr, "Uso: %s primer_sumando segundo_sumando\n", argv[0]);
     return 1;
    }
  sd = socket(PAF_INET, SOCK_STREAM, IPPROTO_TCP);

  bzero((char *)&server_addr, sizeof(server_addr));
  hp = gethostbyname ("localhost");

  memcpy (&(server_addr.sin_addr), hp->h_addr, hp->h_length);
  server_addr.sin_family = AF_INET;
  server_addr.sin_port = htons(56789);
  /* se establece la conexión */
  connect(sd, (struct sockaddr *) &server_addr, sizeof(server_addr));

  num[0]=htonl(atoi(argv[1]));
  num[1]=htonl(atoi(argv[2]));

  write(sd, (char *) num, 2 *sizeof(long int));     /*envía la petición*/

  for(i=0;
      i<sizeof(long int)&&(c=read(sd,((char *)&res)+i,sizeof(long int)-i))>0;i+=c);
   /* recibe la respuesta */

  printf("El resultado es: %d \n", ntohl(res));
  close (sd);
  return 0;
}
```

Las siguentes 4 cuestiones se plantean en el contexto de la sesión FTP que se lleva a cabo, parte de cuya captura se muestra en la imagen. Responder a las cuestiones con Verdadero o Falso. En la línea 40 de la captura, lo que está ocurriendo es:



3. (**0.1 puntos**) El cliente está aceptando un cierre de conexión por parte del servidor.

4. (**0.1 puntos**) El servidor está aceptando un cierre de conexión por parte del cliente.

5. (**0.1 puntos**) El cliente está indicando un cierre de conexión al servidor.

6. (**0.1 puntos**) El servidor está indicando un cierre de conexión al cliente.

7. (**0.1 puntos**) En la captura anterior, ¿a qué línea está respondiendo la línea 41?