Exercise 1 Consider the following source codes of both client and server programs:

Client:

```
int main(int argc, char* argv[]) {
  int sockd;
  struct sockaddr_in serv_name;
  struct hostent *hp;
  int err;

  if (argc < 3) {
    fprintf(stderr, "Usage: %s IP file_name\n", argv[0]);
    exit(1);
  }
   // en argv[1]  server IP
   // en argv[2]  file name of the file to transfer

  sockd = socket(AF_INET, SOCK_STREAM, 0);
  if (sockd == -1) {
    perror("Error socket");
    exit(1);
  }

  bzero((char *)&serv_name, sizeof(serv_name));
  hp = gethostbyname (argv[1]);
  memcpy (&(serv_name.sin_addr), hp->h_addr, hp->h_length);
  serv_name.sin_family = AF_INET;
  serv_name.sin_port = htons(1200);

  status = connect(sockd, (struct sockaddr*)&serv_name, sizeof(serv_name));
  if (err == -1) {
    perror("Server connection error");
    exit(1);
  }

  /* * * * * * * * * CLIENT  * * * * * * * * * * * * * *
    FILL OUT
   */
  exit(0);
}
```

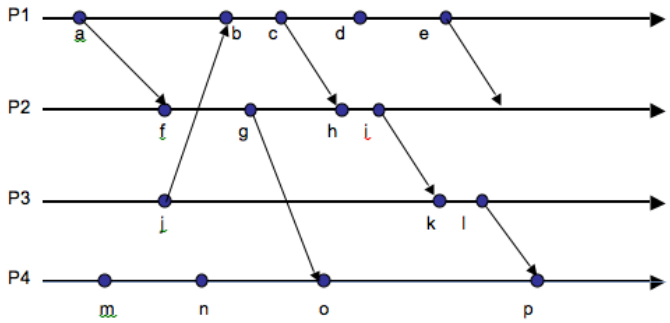
Server:

```
int main() {
  int sockd,sockd2;
  struct sockaddr_in dir, cliente;
  int err, len;

  sockd = socket(AF_INET, SOCK_STREAM, 0);
  if (sockd == -1){
    perror("Socket error");
    exit(1);
  }

  dir.sin_family = AF_INET;
  dir.sin_addr.s_addr = INADDR_ANY;
  dir.sin_port = htons(1200);
  err = bind(sockd, (struct sockaddr*)&dir, sizeof(dir));
  if (err == -1){
    perror("Error en bind");
    exit(1);
  }

  err = listen(sockd, 5);
  if (err == -1){
    perror("Listen error");
    exit(1);
  }

  len = sizeof(peer_name);
  sockd2 = accept(sockd, (struct sockaddr*)&client, &len);
  if (sockd2 == -1) {
      perror("Accept error");
      exit(1);
  }
  /* * * * * * * * * SERVER * * * * * * * * * * * * * *
    FILL OUT
   */
  exit(0);
}
```

The client program must transfer the file content to the server. The file name is obtained by the client on the command line argument argv[2]. The server must create a file with the same name and write file content sent by the client. State:

a) What is the port assigned to the server?
b) What is the port assigned to the client? When the port is assigned to the client?
c) Implement both client and server sides needed to transfer a file from client to server side. The server must create and write the contents of the file received from the client in a file with the same name. (`sendfile` primitive is not allowed).

Exercise 2 The diagram below shows a series of events that occur between four processes of a distributed application.



State:

a) Assigning values of the vector clocks, in the form (v1, v2, v3, v4) to the following graphic events.
b) For the next tuples of events explain if the relationship is $\rightarrow$ or $||$ between them:
   - (j,o)
   - (j,p)

   Justify your answer.

Exercise 3. We want to implement a fire management service by using a distributed client-server architecture. This service has a control unit, a sensor that gathers information on the ambient temperature, a fire sprinkler system and an actuator connected to an alarm button. The unit requires the sensor temperature and humidity recorded every minute. In case the centralized detects a temperature greater than 60 degrees, the unit sends an activation request to the fire sprinkler system. When the sensor temperature drops 30 degrees, the control unit sends a request to stop anti-fire system. If at any time a user presses the alarm button, the actuator that controls this button sends a message to the control panel indicating a fire. Similarly, the unit sends an activation request to fire sprinkler system. When the sensor temperature drops 30 degrees, the control unit sends a request to stop anti-fire system.

1. Make a complete design of the distributed application using sockets, describing in detail the application protocol. Make all considerations considered relevant.
2. Specify the structure using pseudocode for each of the processes running in system. Indicate the sockets library class used given your design.

Exercise 4. A client application generates a request to a web server of 1 MB. If packets sent over the network have more than 64 KB calculate:

a. Number of packets that the client application needs to send to the server.
b. Transmission time of a message assuming that the bandwidth of the network is 100 Mbps and latency is 1 ms per package.
c. Total time to send the complete request.
d. Whereas the connected server in this network has sufficient resources, how many transfers of 1MB could perform in one second?