IC221 Homework: Socket Programming (1) Name \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

AY 22S, 100 points total

(24) For each of the following networking calls, provide a brief, one sentence description:

|  |  |
| --- | --- |
| Call | Description |
| socket() | Opens a socket |
| bind() | Binds server sockets to socket addresses |
| accept() | Accepts new connections and returns a duplicate of he server socket |
| listen() | Listens for incoming connections after being bound |
| connect() | Allows a client to connect to a server and read and write to it |
| close() | Closes the socket |

(15) For each of the arguments to the socket() call below, briefly explain their meaning for the socket being opened.

int sock;

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

|  |  |
| --- | --- |
| Argument | Meaning |
| AF\_INET | Means it is using IPv4 |
| SOCK\_STREAM | Means it is using TCP protocol |
| 0 | Means we are not including extra info about the protocol |

(10) Below is output from the hello\_server.c program in the Networks 4 notes. When the network packet with howdy is sent *from* the client, *to* the server, what are the source and destination port values in that network packet, according to the ouput?

$ ./hello\_server

Listening On: 127.0.0.1:1845

Connection From: 127.0.0.1:60472 (4)

Connection To: 127.0.0.1:1845

Read from client: howdy

Sending: Hello 127.0.0.1:60472

Go Navy! Beat Army

Closing socket

|  |
| --- |
|  |

(20) For each of the arguments and return value in the connect() call below, briefly explain their meaning for the connection being established.

int res;

struct sockaddr\_in \* saddr\_in;

res = connect(sock, (struct sockaddr \*) saddr\_in,   
 sizeof(struct sockaddr\_in));

|  |  |
| --- | --- |
| Element | Meaning |
| sock |  |
| saddr\_in |  |
| sizeof(saddr\_in) |  |
| res |  |

(5) Explain why when *accepting* an *incoming* connection, a *new* socket is created and returned (instead of using the existing one).

|  |
| --- |
|  |

(5) Explain the second argument to listen(), the *backlog*. What does this argument refer to and what does it mean?

|  |
| --- |
|  |

(10) Consider the server code loop below, for handling client socket connections.

|  |
| --- |
| char buf[BUF\_SIZE];  int sockets[NUMSOCKS], i,n;  //iterate over all open sockets  for(i=0;i < NUMSOCKS; i++){  if(i>0){  // read from socket  n = read(sockets[i], buf, BUF\_SIZE);  // socket closed  if(n < 0){  close(sockets[i]);  sockets[i] = -1;  }  // echo back  write(sockets[i], buf, n);  }  } |

Can this server program handle multiple clients simultaneously? That is, if multiple clients are connected, will the server be able to service all the sockets, when data is available from some clients, but not from others? Explain.

|  |
| --- |
|  |