CECS 472, Fall 2012, Midterm 1, Dennis Volper Name:________________________________

Open Book, open notes. All questions equal weight.
Reassemble loops must be shown in every case where one is needed. All numbers sent over the network must be in network standard order. You may assume appropriate includes are already part of the code.
1) Modify the primitive client so that it uses `argv[1]` as the name of the server to connect to.

```
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <string.h>
#include <stdio.h>
int main(int argc; char* argv[]) {
  int s;
  char message[80];
  struct sockaddr_in      srv_addr;
  /* Get a socket. */
  s = socket(PF_INET, SOCK_STREAM, 0);
  /* Put server's address into a socket structure */
  memset(&srv_addr, 0, sizeof(srv_addr));
```

```
  srv_addr.sin_addr.s_addr = htonl(0x868bf811);
    /*134.139.248.17*/
  srv_addr.sin_family = AF_INET;
  srv_addr.sin_port = htons(7654);
  /* Request the connection to the server */
  connect(s, (struct sockaddr *) &srv_addr, sizeof(srv_addr));
  strcpy(message,"Client speaks");
    /*Send a message to server*/
  write(s, message, 80);
    /*Get server's reply*/
  read(s, message, 80);
  close(s);
  return 1;
}
```

2) Write a UDP *gethostname* client. The client will send a message to the server. The server will the hostname of the server (a string). The client will print this information. After printing this information, the client exits. Show only the client procedure, do not show the main program. Be sure to declare any variables you use.

3) Write a UDP `gethostname` server. When the server gets a message from a client it will make a `gethostname` call and send back to the client the value that call returns. Show only the `receivefrom`, the `sendto` and the code that goes in between. Show also the declaration of the buffer variable.

4) Write a TCP *sysinfo* client. The client will connect to the server. The server will send back three pieces of system information. The client will print these three pieces of information (3 integers) After printing this information, the client exits. Show only the client procedure, do not show the main program.

5) Write a TCP `sysinfo` server. When the server gets a connection from a client it will make a `sysinfo` call and send back to the client the `uptime`, `freeram` and `freeswap` values. Show only the service procedure, do not show the main program.