**CHEAT SHEET**

* int socket(int domain, int type, int protocol);
* domain = AF\_INET, AF\_INET6
* type = SOCK\_STREAM, SOCK\_DGRAM
* protocol = 0 (preferred), IPPROTO\_TCP, IPPROTO\_UDP, IPPROTO\_ICMP
* int bind(int socket, struct sockaddr \*name, int namelen)
* struct sockaddr\_in {

short sin\_family; // e.g. AF\_INET, AF\_INET6

unsigned short sin\_port; // e.g. htons(3490)

struct in\_addr sin\_addr; // see struct in\_addr, below

char sin\_zero[8]; // zero this

};

* struct in\_addr {

unsigned long s\_addr; // load with inet\_addr()

};

* struct sockaddr {

unsigned short sa\_family; // address family, AF\_xxx

char sa\_data[14]; // 14 bytes of protocol address

};

* int listen(int socket, int backlog)i
* int accept(int socket, struct sockaddr \*addr, int \*addrlen)
* int connect(int socket, struct sockaddr \*addr, int addrlen)
* int send(int socket, const void \*buf, int buflen, int flags);
* int recv(int socket, void \*buf, int buflen, int flags);
* int sendto(int socket, const void \*buf, int buflen, int flags, struct sockaddr\* to, int tolen);
* int recvfrom(int socket, void \*buf, int buflen, int flags, struct sockaddr\* from, int \*fromlen);
* int close(int socket)

**Multithreading:**

* int pthread\_create(pthread\_t \*thread, pthread\_attr\_t \*attr, void \*(\*start\_routine)(void \*), void \*arg);
* void pthread\_exit(void \*value\_ptr);
* int pthread\_join(pthread\_t thread, void \*\*value\_ptr);

**Filing Syntax:**

* ptr = fopen("filepath","mode")
* int fgetc( FILE \* fp ) //read single char from file
* fclose(fptr)
* type arrayName [ arraySize ]; // To declare an array
* int strcmp(const char \*s1, const char \*s2); // string compare

**Headers File:**

* #include <stdio.h>
* #include <string.h>
* #include <sys/socket.h> //socket
* #include <arpa/inet.h> //inet\_addr
* #include<pthread.h>
* #include<bool.h>