GNU nano 2.0.6 File: server.c

/\* A simple server in the internet domain using TCP

The port number is passed as an argument \*/

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <unistd.h>

#include <sys/types.h>

#include <sys/socket.h>

#include <netinet/in.h>

void error(const char \*msg)

{

perror(msg);

exit(1);

}

#pragma pack(1)

struct sendd

{

int num;

float decnum;

char letter;

};

#pragma pack(0)

int main(int argc, char \*argv[])

{

int sockfd, newsockfd, portno;

socklen\_t clilen;

char buffer[256];

struct sockaddr\_in serv\_addr, cli\_addr;

int n;

if (argc < 2) {

fprintf(stderr,"ERROR, no port provided\n");

exit(1);

}

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

if (sockfd < 0)

error("ERROR opening socket");

bzero((char \*) &serv\_addr, sizeof(serv\_addr));

portno = atoi(argv[1]);

serv\_addr.sin\_family = AF\_INET;

serv\_addr.sin\_addr.s\_addr = INADDR\_ANY;

serv\_addr.sin\_port = htons(portno);

if (bind(sockfd, (struct sockaddr \*) &serv\_addr,

sizeof(serv\_addr)) < 0)

error("ERROR on binding");

listen(sockfd,5);

clilen = sizeof(cli\_addr);

newsockfd = accept(sockfd,

(struct sockaddr \*) &cli\_addr,

&clilen);

if (newsockfd < 0)

error("ERROR on accept");

bzero(buffer,256);

struct sendd send1;

n = read(newsockfd,&send1,sizeof(send1));

send1.num = send1.num \*2;

send1.decnum = send1.decnum + 1;

if (send1.letter >= 'a' && send1.letter < 'z')

{

send1.letter = send1.letter + 1;

}

else if (send1.letter == 'z')

{

send1.letter = 'a';

}

if (n < 0) error("ERROR reading from socket");

printf("Here is the message:\n Double num: %d\n One Plus Float: %.2f\n Next letter: $

n = write(newsockfd,"I got your message",18);

if (n < 0) error("ERROR writing to socket");

close(newsockfd);

close(sockfd);

return 0;

}