**IOT 프로그래밍 1차 과제**

20150327 이진수

/\*실행단계에서 인자값을 받을 수 있게 main함수에 인자값을 넣어줌\*/

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

…(변수 초기화)

/\*socket을 생성하는 부분\*/

s=socket(PF\_INET, SOCK\_STREAM, 0); //TCP통신을 하며 IPV4를 지원하기 위한 인자들을 넣어 생성

…(에러검출부분)

server\_addr.sin\_family=PF\_INET; //프로토콜 도메인을 넣어줌

/\*host의 포트번호 -> network를 통해서 가기 때문에 big endian으로 바꿈\*/

server\_addr.sin\_port=htons(80);

inet\_aton(argc[1], &server\_addr.sin\_addr);

memset(&(server\_addr.sin\_zero),0,8);

/\*주어진 IP를 통해서 만들어놓은 소켓을 server와 연결\*/

if(connect(s, (struct sockaddr \*)&server\_addr, sizeof(server\_addr)) ==-1){

perror("connect");

return 0;

}

/\*socket에 server에게 전하려고했던 명령어 자체를 집어넣음\*/

len=write(s, "GET / HTTP/1.1\r\nHost: 127.0.0.1\r\nConnection: close\r\n\r\n", strlen("GET / HTTP/1.1\r\nHost: 127.0.0.1\r\nConnection: close\r\n\r\n"));

…(에러 검출부분)

/\*server에서 받아오는 값을 화면에 출력\*/

while(1){

read\_len=read(s,msg,1000);

if(read\_len ==-1){

perror("read");

return 0;

}

if(read\_len==0){

printf("\nEND\n");

break;

}

msg[read\_len]=0;

printf("%s",msg);

}

/\*close socket\*/

close(s);

return 0;

}