/\*

\* lookup9 : does no looking up locally, but instead asks

\* a server for the answer. Communication is by Internet UDP Sockets

\* The name of the server is passed as resource. PORT is defined in dict.h

\*/

#include <sys/types.h>

#include <sys/socket.h>

#include <string.h>

#include <netdb.h>

#include <netinet/in.h>

#include "dict.h"

int lookup(Dictrec \* sought, const char \* resource) {

static int sockfd;

static struct sockaddr\_in server;

struct hostent \*host;

static int first\_time = 1;

if (first\_time) { /\* Set up server address & create local UDP socket \*/

first\_time = 0;

/\* Set up destination address.

\* Fill in code. \*/

memset(&server,'\0',sizeof(server));

server.sin\_family = AF\_INET;

server.sin\_port = htons(PORT);

if((host = gethostbyname(resource))==NULL){

perror("gethostbyname");

exit(1);

}

//int h\_addrtype; 主機IP地址的類型

memcpy((char \*)&server.sin\_addr, host->h\_addr,host->h\_length);//int h\_length; 主機ip地址長度

/\* Allocate a socket.

\* Fill in code. \*/

if((sockfd = socket(AF\_INET,SOCK\_DGRAM,0))==-1){

perror("socket");

exit(1);

}

}

/\* Send a datagram & await reply

\* Fill in code. \*/

int len = sizeof(struct sockaddr\_in);

sendto(sockfd,sought,sizeof(Dictrec),0,(struct sockaddr \*)&server,len);

recvfrom(sockfd,sought,sizeof(Dictrec),0,(struct sockaddr \*)&server,&len);

if (strcmp(sought->text,"XXXX") != 0) {

return FOUND;

}

return NOTFOUND;

}