/\*

\* msgq\_server : serve clients using a message queue.

\* rcv.content.id contains the type of the message:

\* type 1 messages are for the server

\* type n messages are for client whose pid is n

\* argv[1] is the name of the local file

\* argv[2] is the message-queue key

\* To convert from string to long, use strtol, since

\* it copes sensibly with numbers like 0xff

\*/

#include <stdlib.h>

#include <string.h>

#include <sys/stat.h>

#include <sys/ipc.h>

#include <sys/msg.h>

#include <errno.h>

#include <sys/types.h>

#include "dict.h"

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

int qid;

Dictrec tryit;

static ClientMessage rcv;

ServerMessage snd;

struct stat stbuff;

if (argc != 3) {

fprintf(stderr,"Usage : %s <dictionary source>"

" <resource / message queue key>\n",argv[0]);

exit(errno);

}

/\* Cannot find local database file. \*/

if (stat(argv[1],&stbuff) == -1) {

DIE(argv[1]);

}

/\* Get the message queue, key is based on commandline second argument.

\*

\* Fill in code. \*/

key\_t key;

if(( key = strtol (argv[2],NULL,16))==0){DIE("strtol");}

if((qid = msgget(key, IPC\_CREAT | 0600)) == -1){DIE("msgget");}

for (;;) { /\* await client messages ; reply immediately \*/

/\* Wait for / receive a message.

\*

\* Fill in code. \*/

if(msgrcv(qid, &rcv, sizeof(ClientMessage)-sizeof(long), 1L, 0) == -1) {DIE("msgrcv");}

strcpy(tryit.word,rcv.content.word);/\* Get the word to lookup. \*/

snd.type = atol(rcv.content.id); /\* Get sender to set msg type.\*/

switch(lookup(&tryit,argv[1])) { /\* Lookup word in db. \*/

case FOUND:

strcpy(snd.text,tryit.text); /\* Found. Put result in return msg. \*/

break;

case NOTFOUND : /\* Not found. Return XXXX. \*/

strcpy(snd.text,"XXXX");

break;

case UNAVAIL : DIE(argv[1]); /\* Other problem. \*/

}

/\* Send response.

\*

\* Fill in code. \*/

if(msgsnd(qid,&snd,sizeof(ServerMessage)-sizeof(long),0) == -1){DIE("msgsnd");}

}

}