/\*

\* redirect\_in.c : check for <

\*/

#include <unistd.h>

#include <fcntl.h>

#include <errno.h>

#include <string.h>

#include <stdio.h>

#include <stdlib.h>

#include "shell.h"

#define STD\_OUTPUT 1

#define STD\_INPUT 0

/\*

\* Look for "<" in myArgv, then redirect input to the file.

\* Returns 0 on success, sets errno and returns -1 on error.

\*/

//sort -i < mess

int redirect\_in(char \*\* myArgv) {

int i = 0;

int fd; //文件描述符

/\* search forward for <

\*

\* Fill in code. \*/

int number=1;

for(i=0;myArgv[i]!=NULL;i++){

if(myArgv[i] == NULL) break;

number = strcmp(myArgv[i],"<");

if(number == 0) break;

}

if (myArgv[i]) { /\* found "<" in vector. \*/

/\* 1) Open file.

\* 2) Redirect stdin to use file for input.

\* 3) Cleanup / close unneeded file descriptors.

\* 4) Remove the "<" and the filename from myArgv.

\*

\* Fill in code. \*/

/\* Open file \*/

fd = open(myArgv[i+1],O\_RDWR|O\_CREAT|O\_APPEND,0644);//EX:open cat 123<mess =myArgv[0] myArgv[1] myArgv[2] myArgv[3]

if( fd < 0){ //可讀的寫的，wr-w--w-- // 06444代表user可以讀可以寫

//myArgv[i+1] => > or < 後面必定是檔名

perror("open");

exit(1);

/\*printf("open error [%d]\n",errno); //return -1 代表開啟失敗

\*return -1;

\*/

}

/\* Redirect stdin to use file for input \*/

dup2(fd,STD\_INPUT); //0 stdin 標準輸入 ,1 stdout 標準輸出 ,2 stderr 標準錯誤\*/原本從鍵盤key，改由當暗輸入

//EX: sort -i < mess;

/\*Cleanup / close unneeded file descriptors\*/

close(fd);

/\*Remove the "<" and the filename from myArgv\*/

myArgv[i]=NULL; //memset(myArgv[i],'\0',1);

}

return 0;

}