**今天的工作**

继续完成功能，从完成管道和文件重定向功能开始

管道：比如ls | grep main.c（在当前目录下找到main.c后缀的文件）

**遇到的问题**

1. 怎么实现pipe
   1. 使用什么exec()族函数

**解决的问题**

1. 使用execvp函数
   1. 如果成功执行没有返回值，发生异常返回-1
   2. 第一个参数是要执行的指令，第二个参数是参数列表

比如我要执行的是cmd[0]中存放的文件(从环境变量PATH开始查找)，参数列表就是要传入的参数；列表。

execvp(cmd[0],cmd);

**关键代码**

**1.getHostname**

#include <stdio.h>

#define max\_name\_len 40

/\*\*\*\*\* Info \*\*\*\*\*/

/\* Author: zhj \*/

/\* Function: get hostname\*/

/\*\*\*\*\* Info \*\*\*\*\*/

void getHostname\_zhj(){

const int max\_name\_len = 256;

extern struct passwd \*pwd;

int uid;

char hostname[max\_name\_len];

gethostname(hostname,max\_name\_len);

uid=geteuid();

printf("%s@%d\n",hostname,uid);

}

int main()

{

getHostname\_zhj();

return 0;

}

**2.getUsername**

/\*\*\*\*\* Info \*\*\*\*\*/

/\* Author: zhj \*/

/\* Function: 得到用户名\*/

/\*\*\*\*\* Info \*\*\*\*\*/

#include <stdio.h>

#include <string.h>

#include <pwd.h>

void getUsername\_zhj(){

/\* Note： max\_name\_len是用户名最大的长度\*/

/\*这里定义为40\*/

const max\_name\_len=40;

char username[max\_name\_len];

struct passwd\* pwd = getpwuid(getuid());

strcpy(username, pwd->pw\_name);

printf("%s\n",username);

}

int main()

{

getUsername\_zhj();

return 0;

}

**其他想法**

无

**明天的计划**

继续完成功能，我打算从完成管道和文件重定向功能开始

管道：比如ls | grep main.c（在当前目录下找到main.c后缀的文件）