

深圳大学期末考试试卷

开/闭卷 开卷

A/B 卷

课程编号 1503020001

课序号 01

课程名称 系统编程

学分 3

命题人(签字) 审题人(签字) 年 月 日

题号	一	二	三	四	五	六	七	八	九	十	基本题 总分	附加题
得分												
评卷人												

特别说明：所有作答过程建议在华为 Taishan 服务器+openEuler 上完成。

在作答之前，请首先确认你的实验环境满足以下要求：

1. 请以名为“学生姓名拼音+学号”的用户登录、完成作业，如学生名张三，学号 123456，则用户名为 zhangsan123456。请特别注意，以用户名 zhangsan123456 完成的作业记为 0 分。

2. 用户提示符需要包含用户名、主机名和工作目录。

参考样例 1) 名为 szu 的用户在她/他的虚拟机 taishan02-vm-10 下工作时，工作目录为 process，则终端提示符显示为（此时，环境变量 PS1='[\u@\h\W]\\$ '）：

```
[szu@taishan02-vm-10 process]$
```

参考样例 2) 用户 caichengru2020152021 在虚拟机 taishan01-sp-2023-3 的主工作（~）时，终端提示符显示为

```
caichenru2020152021 at taishan01-sp-2023-3 in ~
```

3. 请按题目要求完成编码和测试，按要求运行代码、截屏，填写 **2023 年度《系统编程》大作业答题纸**，在截止日期（**2024 年 1 月 2 日 23:30**）前通过 Blackboard 提交**答题纸电子版和代码**，2024 年 1 月 3 日 17:30 前提

交纸质版本。没有赶上 Blackboard 提交的截止日期的同学，可通过 QQ、微信、电子邮件等方式联系主讲老师提交电子版，并在 2023 年 1 月 8 日 17:30 前提交纸质版，这之后不再接收任何答卷。特别注意：2024 年 1 月 2 日 23:30~2024 年 1 月 8 日之间提交电子版大作业的计入迟交，迟交的同学期末总评最高分为 92，即不会有 A+的机会。

4. 必须提交纸质答卷，两种方式：1) 平时课堂上提交，特别注意的是，最后一次课是 2024 年 1 月 2 日下午，致腾楼 241。2) 主讲老师办公室：粤海校区小西门下文山湖湖边粤海门广场实验楼 A 座 202。
5. 大作业分数的评定将根据提交的答题纸上实验结果截屏的正确性、可读性、描述与分析有效性和逻辑性、代码的质量等。所有截屏终端提示符信息与学生姓名学号不符的皆为 0 分。
6. 本次考核设置 2 道大题，任选 1 题作答即可。请根据个人兴趣与能力选答。期末得分与选题无关，开源贡献看贡献质量，A+需要有功能增强。所有开源贡献需要有网站或开源贡献收录邮件截图证明。

一、 利用操作系统提供的系统调用、标准C库函数和POSIX 线程库pthread 函数设计并实现多人即时聊天软件。并利用学过的知识深入分析问题

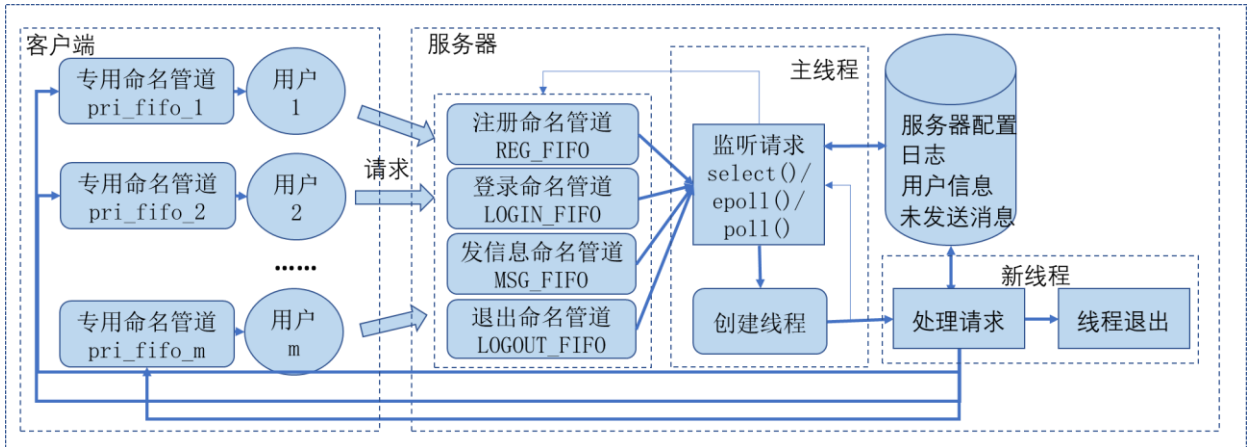


图 1. 单机多用户聊天系统参考架构图

1. 服务器初始化（40 分），见图 1，服务器是多线程守护进程。

1.1. 守护进程命名规则：chatserver_学生姓名拼音首字母缩写，例如，张三的守护进程命名为：chatserver_zs。

1.2. 服务器配置文件设置以下信息：

1.2.1. 众所周知的命名管道

- a) REG_FIFO : 命名为“学生姓名拼音首字母缩写+register”
- b) LOGIN_FIFO : 命名为“学生姓名拼音首字母缩写+login”
- c) MSG_FIFO : 命名为“学生姓名拼音首字母缩写+sendmsg”
- d) LOGOUT_FIFO : 命名为“学生姓名拼音首字母缩写+logout”

1.2.2. 同时在线用户数最大值 MAX_ONLINE_USERS，该值为 4。同一用户同一时刻只允许在一个终端登录一次 MAX_LOG_PERUSER, 该值为 1。

1.2.3. 用户日志文件存放目录 LOGFILES: /var/log/chat-logs/

1.3. 服务器首先读取配置文件，通过配置文件约束行为。

1.3.1. 根据配置文件创建众所周知的命名管道，采用多路复用技术（select()、epoll()或poll()）监听并读取以上信息，分别启动线程处理用户的注册请求、登录请求、聊天请求和退出请求，并将结果返回给用

户，并做日志记录。服务器要求运行期间忽略 SIGCHLD、SIGTERM 等信号。

1.3.2. 为每个注册成功的用户建立独立日志文件，统一放在 1.2.3 设置的用户日志文件存放目录 LOGFILES 下。日志文件收集用户注册事件（用户名，注册，时间）、登录事件（用户名，登录，时间）、退出事件（用户名，退出，时间）、成功发送的信息（发送者，接收者，时间，true），未成功发送的信息（发送者，接收者，时间，false），日志文件只有超级用户可以读、写。同组用户和其他用户一律取消读、写、运行等权限。如果有未成功发送的信息，当接收者重新登录时服务器主动推送这些信息，并写相应日志文件（发送者，接收者，时间，true）。为安全起见，未成功发送的信息（发送者，接收者，信息，时间）应该存在特定的地方，更安全地保护起来（难点）。

请在答卷完成以下内容：

- 1) （10 分）请截图显示守护进程相关的进程信息、公共 FIFO 相关的信息，并解释各特征字段含义，截图显示命名管道创建、打开的代码。
 - 2) （10 分）请截图显示配置文件内容及服务器端相应的处理代码，服务器忽略 SIGCHLD、SIGTERM 信号的代码
 - 3) （10 分）请截图显示日志文件生成代码及有效日志文件属性及其内容，包括日志文件存放路径、权限，注册、登录、退出、成功发送消息和未成功发送消息事件记录的日志信息，未成功发送信息的管理等（难点）。
 - 4) （10 分）展示多路复用监听请求的代码、解释运行过程 select()、epoll() 或 poll() 的参数文件描述符集合的变化。特别提醒，代码质量将作为评分点。
2. 基本聊天功能展示（40）：用户可以通过客户端注册、登录、收发信息和退出。用户注册时提供用户名和密码，不允许不同用户使用同一用户名注

册。客户端将为每个用户建立一个以用户名为名字的专用命名管道，用户 A 发送给用户 B 的信息首先发送给服务器，然后服务器通过 B 的专用命名管道发送给用户 B。守护进程各线程通过共享内存共同维护和使用一个在线用户列表，包含用户名、密码及其私有命名管道。请注意其中的互斥访问和线程安全，请截图并讨论你的线程安全实现。

1) **截图用户注册的线程函数**，截屏显示以下过程：启动 6 个终端，分别用以下用户名依次注册：学生姓名拼音_1、学生姓名拼音_2、 学生姓名拼音_3、最后学生姓名拼音_4、学生姓名拼音_5、学生姓名拼音_1。守护进程将注册不成功的信息返回给相应的用户。（8 分）。

2) **截图客户端登录线程函数**，截屏显示以下登录过程：启动 6 个终端，分别用以下用户名登录：学生姓名拼音_1、学生姓名拼音_2、 学生姓名拼音_3、学生姓名拼音_4、学生姓名拼音_5、学生姓名拼音_3。守护进程将成功登录的用户信息及所有在线的总人数及用户名显示给所有的用户，不成功的信息仅返回给不成功的用户端。（12 分）名为“学生姓名拼音_2”的用户退出，守护进程将其退出的信息及剩余在线用户的信息显示给剩余的用户。保持名为“学生姓名拼音_2”用户的退出状态做 3)（4 分）。

3) **截图用户消息发送过程，一对一发送**：“学生姓名拼音_1”的用户给“学生姓名拼音_2”的用户发送信息“`How are you, 学生姓名拼音_2`”，**一对多发送**：“学生姓名拼音_1”的用户给“学生姓名拼音_2”、“学生姓名拼音_3”的用户发送信息“`Hi, let's play badminton?`”，截屏三个用户的收发信息情况，包括成功和**不成功的情况（可能较难）**。最后名为“学生姓名拼音_2”的用户登录系统，收到前面的信息，展示发送的时间。（12 分）。

4) **截图讨论主线程收集新线程退出状态与否的代码实现，并讨论其优**

点和缺点（4分）

3. 基于线程池的多用户聊天服务器设计与实现（20分）（难点）：系统参考架构图见图2，请结合代码介绍线程池的管理。

- 在配置文件里新增线程池规模参数 POOLSIZE；值为4；
- 服务器启动时根据配置文件新建 POOLSIZE 个线程，运行期接收请求时分派线程，请求处理完后回收线程用于再分派。请给出线程池管理数据结构设计、线程池线程状态管理算法设计和实现等，并附上相应的运行展示。线程池中线程被分派和回收的时间记录在日志 /var/log/chat-logs/threads-log。截图展示日志信息。

特别提醒：代码质量和是否按时提交是本题评分点之一，请结合程序的正确性、线程的安全性、用户信息的安全性等讨论，迟交的同学本题分数直接减分，减分范围 0~20 分，以确保总评成绩不为 A+ 为准。即如果该生整体成绩为 A，虽然迟交，不扣分。如果该学生平时成绩基本满分，期末答卷也接近满分，但迟交，则本题将扣至该生不拿 A+ 为止。

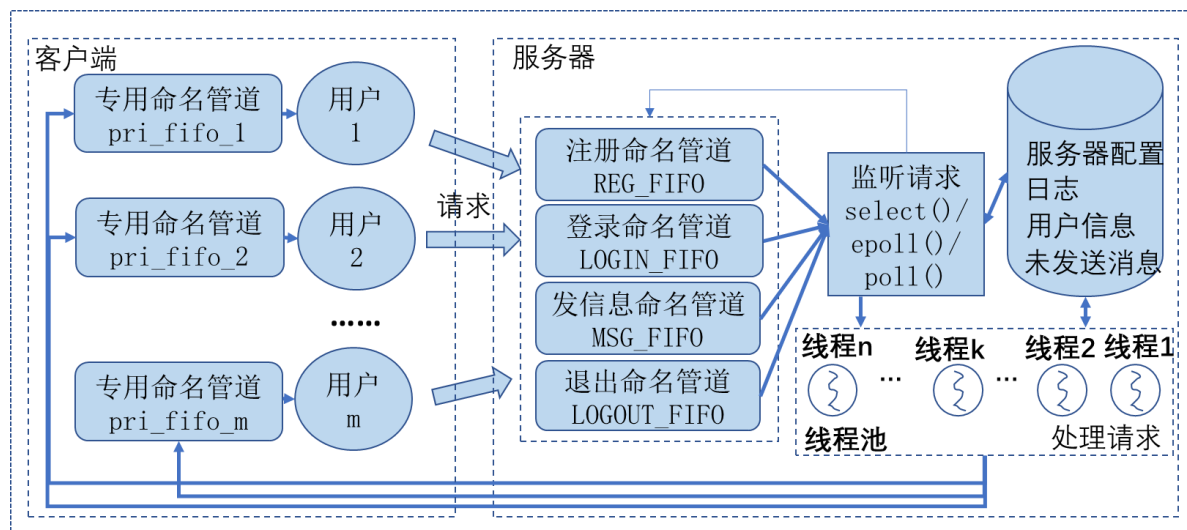


图 2. 基于线程池的单机多用户聊天系统参考架构图

二、（难）openEuler 开源实习官网，领取一项任务并完成，贡献社区后提交任务报告：<https://www.openeuler.org/zh/internship/>），