

《网络与通信》课程实验报告

实验 2: Socket 通信编程

姓名	邱姜铭	院系	计算机学院	学号	22122861
任课教师	刘通	指导教师	刘通		
实验地点	计 706	实验时间	15:00		
实验课表现	出勤、表现得分(10)	实验报告 得分(40)	实验总分		
	操作结果得分(50)				

实验目的:

1. 掌握 Socket 编程过程;

2. 编写简单的网络应用程序。

实验内容:

利用你选择的任何一个编程语言,分别基于 TCP 和 UDP 编写一个简单的 Client/Server 网络应用程序。具体程序要求参见《实验指导书》。

要求以附件形式给出:

● 系统概述:运行环境、编译、使用方法、实现环境、程序文件列表等;

● 主要数据结构;

● 主要算法描述;

● 用户使用手册;

● 程序源代码;

实验要求:(学生对预习要求的回答)(10 分)

得分:

● Socket编程客户端的主要步骤

1. 建立对指定地址端口的Socket

2. 发送数据

3. 等待服务器返回数据

4. 释放Socket

● Socket编程服务器端的主要步骤

1. 建立监听指定端口的Socket

2. 等待请求

3. 处理请求并发送

4. 释放Socket

实验过程中遇到的问题如何解决的?(10 分)

得分:

<p>问题 1：服务端退出时总是无法正常退出 通过排查定位到问题是线程池中线程未停止导致的 最终指向 Socket 一直在等待客户端发来的数据 通过在关闭服务器时 close 所有 socket 解决</p> <p>问题 2：实现 TCP 服务器后实现 UDP 服务器时发现代码结构不兼容 通过比较两种协议通信方式的异同点后重构代码解决</p> <p>问题 3：退出程序时会导致 Socket 抛出 IO 异常导致打印错误信息 通过判断 isRunning 标志位来忽略停止运行后的异常</p>	
本次实验的体会（结论）（10 分）	得分：
<p>1. Socket 通信的灵活性与挑战：</p> <ul style="list-style-type: none">• 实现了 TCP 和 UDP 双协议的支持，TCP 提供了可靠的通信，UDP 则适用于高效快速的场景，但没有可靠性保障。• 同时使用 TCP 和 UDP 的监听机制，保证服务器可以处理多种通信需求，这对设计接口的灵活性要求较高。 <p>2. 线程安全和多客户端通信：</p> <ul style="list-style-type: none">• 服务端处理多客户端连接时，必须考虑线程安全问题，特别是广播功能。通过使用 Java 的线程池和线程安全集合（如 ConcurrentHashMap）确保多个客户端并发情况下的通信稳定性。• 服务端同时支持 TCP 和 UDP 多客户端通信，这需要设计合理的架构来确保不同协议之间的通信不会相互干扰。 <p>3. Echo 与广播功能的实现：</p> <ul style="list-style-type: none">• Echo 功能的实现测试了基本的客户端-服务器通信。服务端能够准确无误地返回客户端发送的消息，表明了消息的接收与发送流程的正确性。• 广播功能通过将接收到的消息传递给所有在线客户端，进一步增加了服务端的复杂性。这也体现了处理多个客户端通信时，服务端消息分发的效率和正确性。 <p>4. Java Socket 编程实践的深入理解：</p> <ul style="list-style-type: none">• 通过这次实验，对 Java Socket 编程有了更深入的理解，包括底层的通信原理、网络协议的选择以及并发通信的处理。• 面对多种复杂的网络环境，合理选择并实现不同的通信协议，是网络编程中的一个核心挑战。 <p>5. 代码模块化与可扩展性：</p> <ul style="list-style-type: none">• 项目结构清晰，通过模块化的方式组织代码，确保了各个部分功能的独立性和可维护性。客户端和服务端的处理逻辑分离，便于后续扩展新功能。 <p>通过本次实验，进一步提升了网络编程技能，尤其是在处理并发通信、网络协议选择、用户管理与安全性等方面的实践经验。</p>	

思考题：（10 分）	
思考题 1：（4 分）	得分：
<p>你所用的编程语言在 Socket 通信中用到的主要类及其主要作用。</p> <p>Socket: tcp 通信时建立的 Socket</p> <p>ServerSocket: tcp 服务器监听时需要的类</p> <p>DatagramSocket: udp 通信时建立的 Socket</p> <p>DatagramPacket: udp 通信时发送的数据包</p>	
思考题 2：（6 分）	得分：
<p>说明 TCP 和 UDP 编程的主要差异和特点。</p> <p>TCP 需要建立连接才能发送消息，同时服务端需要监听端口并并行的创建连接，而且服务端可以通过连接是否中断判断客户端是否在线。</p> <p>UDP 只要向指定地址发送数据包就可以了，服务端是串行的处理请求，同时不能通过连接状态判断客户端是否在线。</p>	
指导教师评语：	
<div>日期：</div>	