**华中科技大学计算机学院**

**《计算机通信与网络》实验报告**

姓名 梅朝瑞 班级 CS1609 学号 U201614758

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 项目 | Socket编程  （30%） | 数据可靠传输协议设计  （15%） | CPT组网  （15%） | 实验报告  （20%） | 平时成绩(20%) | 总分 |
| 得分 |  |  |  |  |  |  |

教师评语：

教师签名：

给分日期：

**目 录**

[实验一 Socket编程实验 1](#_Toc530570686)

[**1.1 环境** 1](#_Toc530570687)

[**1.2 系统功能需求** 1](#_Toc530570688)

[**1.3 系统设计** 1](#_Toc530570689)

[**1.4 系统实现** 2](#_Toc530570690)

[**1.5 系统测试及结果说明** 2](#_Toc530570691)

[**1.6 其它需要说明的问题** 5](#_Toc530570692)

[实验二 数据可靠传输协议设计实验 6](#_Toc530570693)

[**2.1 环境** 6](#_Toc530570694)

[**2.2 实验要求** 6](#_Toc530570695)

[**2.3 协议的设计、验证及结果分析** 6](#_Toc530570696)

[**2.4 其它需要说明的问题** 11](#_Toc530570697)

[实验三 基于CPT的组网实验 12](#_Toc530570698)

[**3.1 环境** 12](#_Toc530570699)

[**3.2 实验要求** 12](#_Toc530570700)

[**3.3 基本部分实验步骤说明及结果分析** 14](#_Toc530570701)

[**3.4 综合部分实验设计、实验步骤及结果分析** 20](#_Toc530570702)

[**3.5 其它需要说明的问题** 23](#_Toc530570703)

[心得体会与建议 24](#_Toc530570704)

[**4.1 心得体会** 24](#_Toc530570705)

[**4.2 建议** 24](#_Toc530570706)

# 实验一 Socket编程实验

## **1.1 环境**

### **1.1.1 开发平台**

（开发机器的硬件配置、系统软件组件、开发平台、使用的第三方组件）

AMD Ryzen 5 1600 Six-Core Processor 3.20 GHz

Windows 10 专业版 Build 17763.195

Microsoft Visual C++ 2017

### **1.1.2 运行平台**

AMD Ryzen 5 1600 Six-Core Processor 3.20 GHz

Windows 10 专业版 Build 17763.195

## **1.2 系统功能需求**

* 必须基于Socket编程，不能直接借用任何现成的组件。
* 提交实验设计报告和源代码；实验设计报告必须包括程序流程图，源代码必须加详细注释。
* 实验设计报告需提交纸质档和电子档，源代码、编译说明需提交电子档。
* 基于自己的实验设计报告，通过实验课的上机试验，将源代码编译成功，运行演示给实验指导教师检查。

编写一个支持多线程处理的Web服务器软件，要求如下：

**第一级：**

* 可配置Web服务器的监听地址、监听端口和虚拟路径。
* 能够单线程处理一个请求。当一个客户（浏览器,输入URL：http://127.0.0.1/index.html）连接时创建一个连接套接字；
* 从连接套接字接收http请求报文，并根据请求报文的确定用户请求的网页文件；
* 从服务器的文件系统获得请求的文件。 创建一个由请求的文件组成的http响应报文。（报文包含状态行+实体体）。
* 经TCP连接向请求的浏览器发送响应，浏览器可以正确显示网页的内容；
* 服务可以启动和关闭。

**第二级：**

* 支持多线程，能够针对每一个新的请求创建新的线程，每个客户请求启动一个线程为该客户服务；
* 在服务器端的屏幕上输出每一个请求的来源（IP地址、端口号和HTTP请求命令行）
* 支持一定的异常情况处理能力。

**第三级：**

* 能够传输包含多媒体（如图片）的网页给客户端，并能在客户端正确显示；
* 对于无法成功定位文件的请求，根据错误原因，作相应错误提示。
* 在服务器端的屏幕上能够输出对每一个请求处理的结果。
* 具备完成所需功能的基本图形用户界面（GUI），并具友好性

## **1.3 系统设计**

系统中包含最为主要的server.cpp程序，主页index.html，报错使用的404.html，主页引用的资源HTML、css、images文件夹，以及数个用作单独测试的.jpg图片文件。

server.cpp中应引用winsock2.h进行编程，包含winsock初始化部分、监听socket创建和绑定部分、多线程工作部分、接收到客户端请求后的会话处理部分等。

## **1.4 系统实现**

（说明你是如何实现系统的，包括但不限于技术关键及解决方案、关键功能的流程图）

在server.cpp中，使用winsock库和windows线程池（<thread>库）编程。

程序分为两个函数：main函数用于创建socket，建立与客户端的多线程会话连接；serverclient函数用于在会话中实际处理客户端的请求，即根据接收到报文的内容进行正则表达式匹配解析，提取有效信息并提供相应服务。

基本上，完成对两种对象文件的传输：html文件表示的网页和jpg文件表示的图片。

在寻找到匹配的文件后，通过缓冲区进行传输。

若未找到对应文件，则打开404.html页面进行提示。

## **1.5 系统测试及结果说明**



图1-1 主页index.html

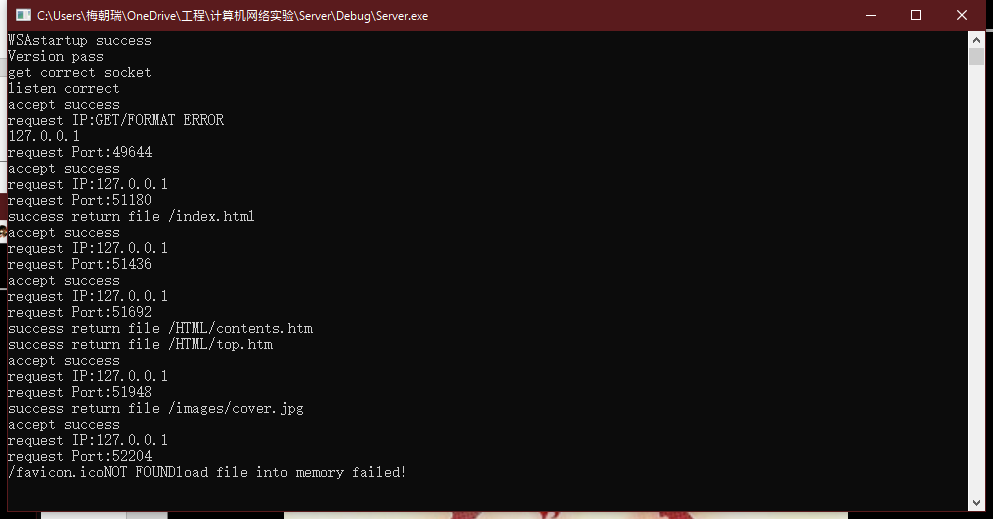


图1-2 请求主页的log

请求主页时，先建立连接，传输index.html。渲染过程中客户端请求index.html中包含的其他资源文件，即contents.htm、top.htm（这两个格式相同，一次性传了过去）、cover.jpg，一一对其进行传输。而最后有favicon.ico，不存在于资源文件夹中，报NOT FOUND。

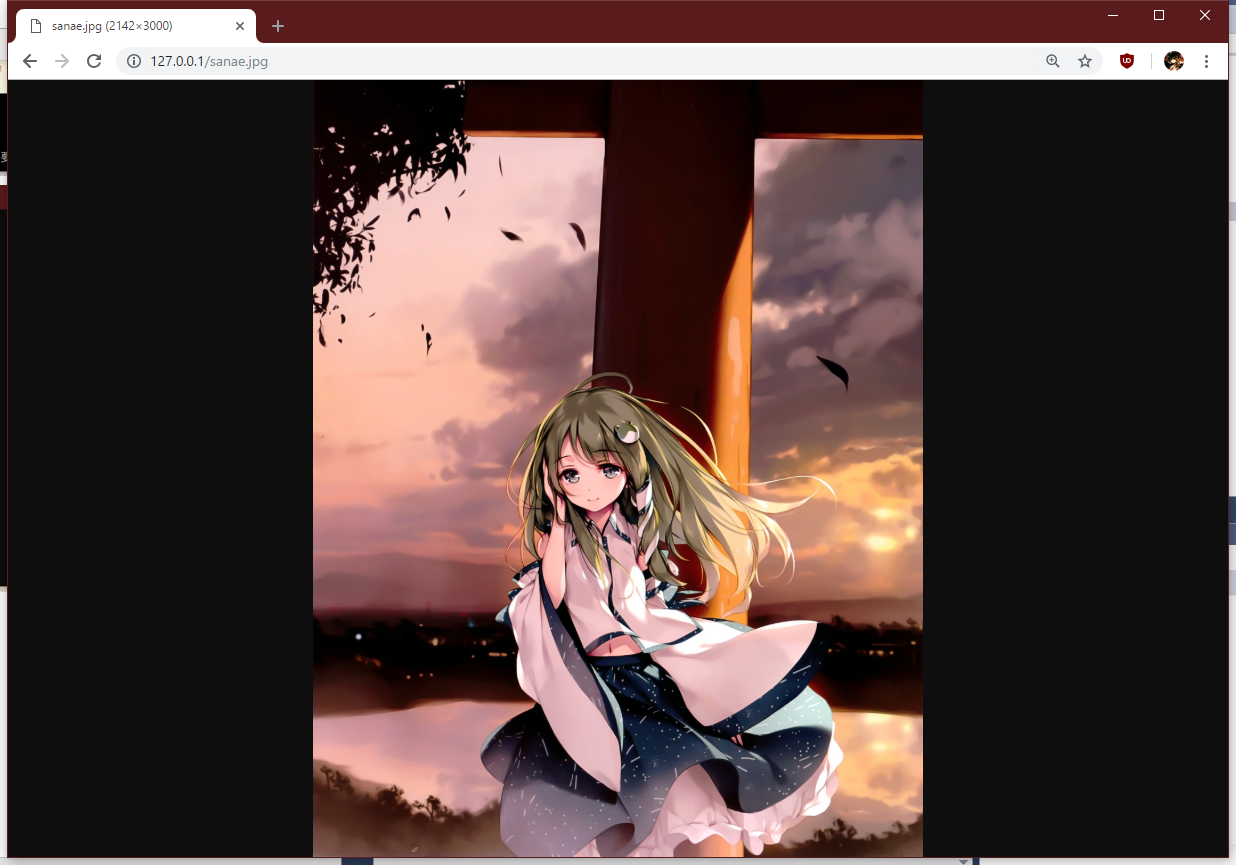


图1-3 图片测试sanae.jpg

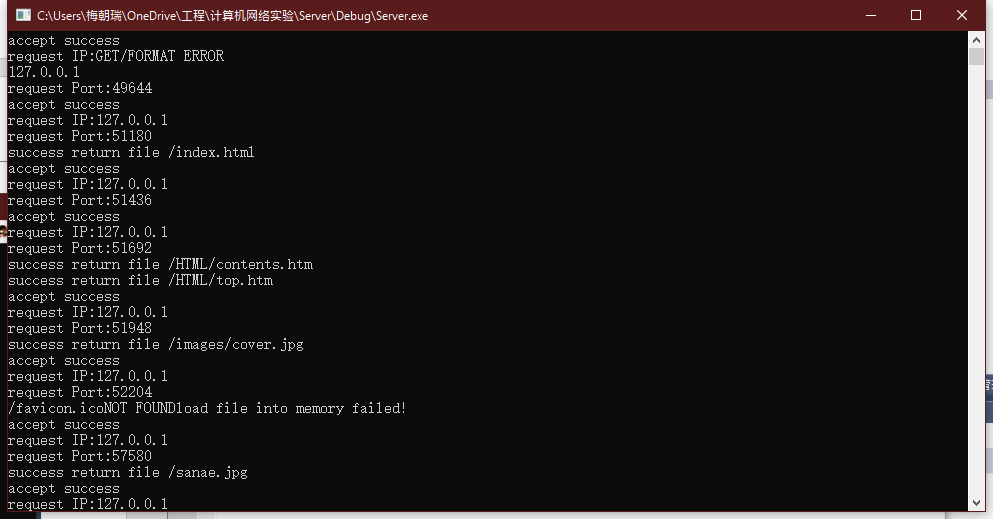


图1-4 请求sanae.jpg的log

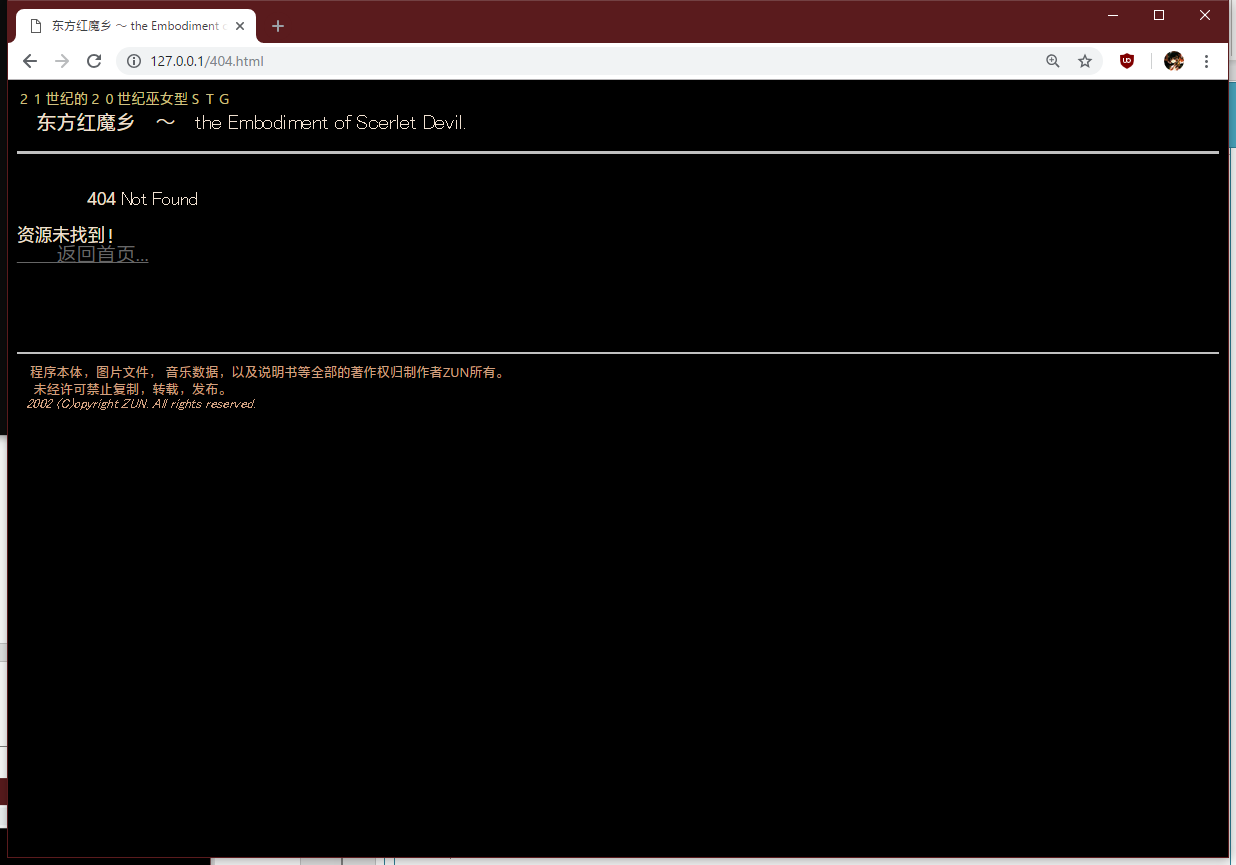


图1-5报错提示页面404.html



图1-6 NOT FOUND错误信息

## **1.6 其它需要说明的问题**

无

# 实验二 数据可靠传输协议设计实验

## **2.1 环境**

AMD Ryzen 5 1600 Six-Core Processor 3.20 GHz

Windows 10 专业版 Build 17763.195

Microsoft Visual C++ 2017

## **2.2 实验要求**

可靠运输层协议实验只考虑单向传输，即：只有发送方发生数据报文，接收方仅仅 接收报文比给出确认报文。

要求实现具体协议时，指定编码报文序号的二进制位数（例如 3 位二进制编码报文 序号）以及窗口大小（例如大小为 4），报文段序号必须按照指定的二进制位数进行编码。

代码实现不需要基于 Socket API，不需要利用多线程，不需要任何 UI 界面

本实验包括三个级别的内容，具体包括：

实现基于 GBN 的可靠传输协议；

实现基于 SR 的可靠传输协议；

在实现 GBN 协议的基础上，根据 TCP 的可靠数据传输机制（包括超时后只重传最早发送且没被确认的报文、快速重传）实现一个简化版的 TCP 协议。报文段格式、报文段序号编码方式和 GBN 协议一样保持不变，不考虑流量控制、拥塞控制，不需要估算 RTT 动态调整定时器 Timeout 参数。

## **2.3 协议的设计、验证及结果分析**

### **2.3.1 GBN协议的设计、验证及结果分析**

开始传输时，启动定时器。每次发送报文前，检查校验和是否正确。

若校验和正确，且确认序号在窗口中，则发送报文，关闭定时器。

若定时器时间到，则需要重发报文。重新启动发送方定时器并重发定时开始时传输的数据包。

即：发送方在发完一个数据帧后，连续发送若干个数据帧，即使在连续发送过程中收到了接收方发来的应答帧，也可以继续发送。且发送方在每发送完一个数据帧时都要设置超时定时器。只要在所设置的超时时间内仍收到确认帧，就要重发相应的数据帧。

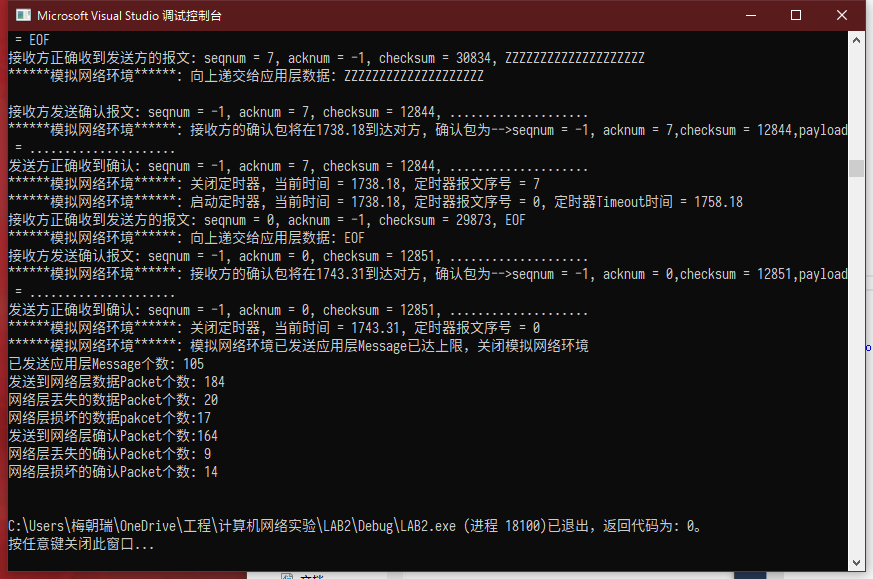


图2-1 GBN单独执行测试

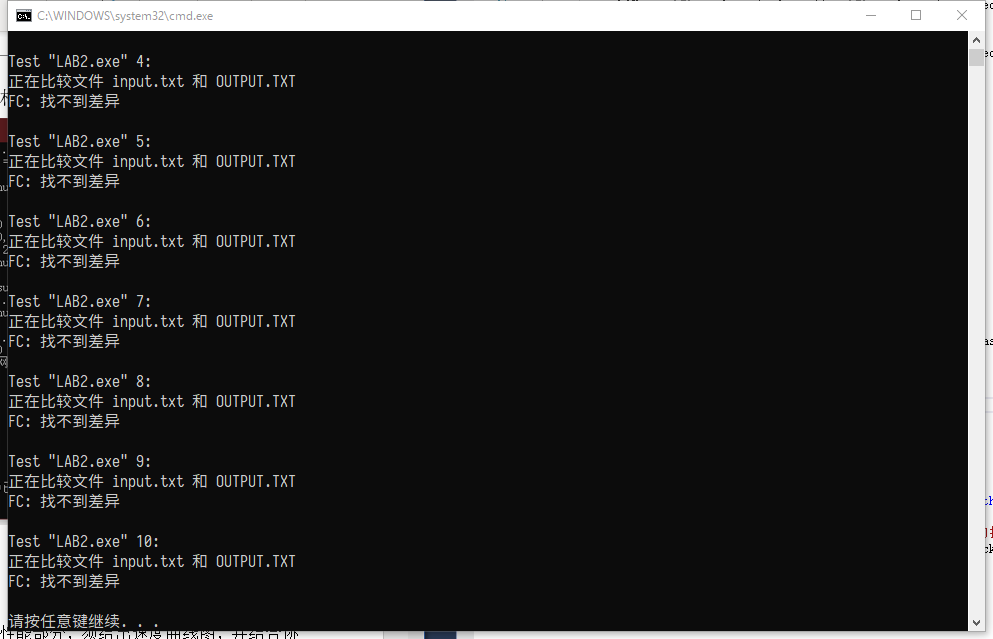


图2-2 check\_win测试GBN

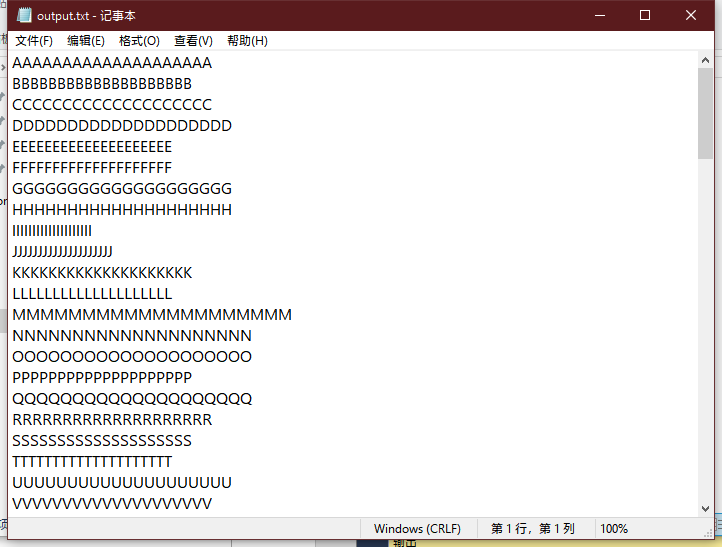


图2-3 测试结果

可以看出，GBN协议成功实现了稳定无误的传输。

### **2.3.2 SR协议的设计、验证及结果分析**

当发送窗口未满时，SR协议为每个传输的分组设置一个定时器，一旦某个分组的定时器到时，则立刻发起重传。确认号ACK只定义完好接收的那一个分组的序号,并不反馈任何其他分组的信息。

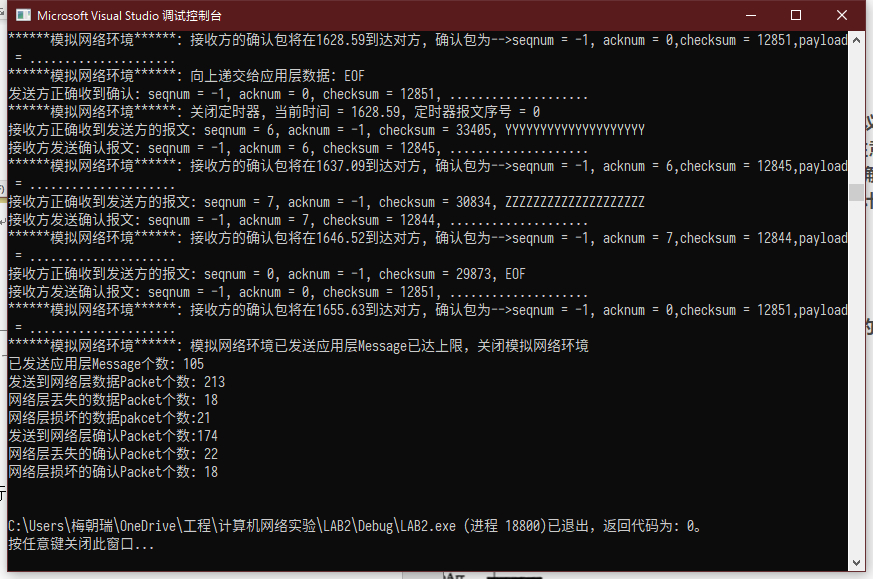


图2-4 SR单独执行测试

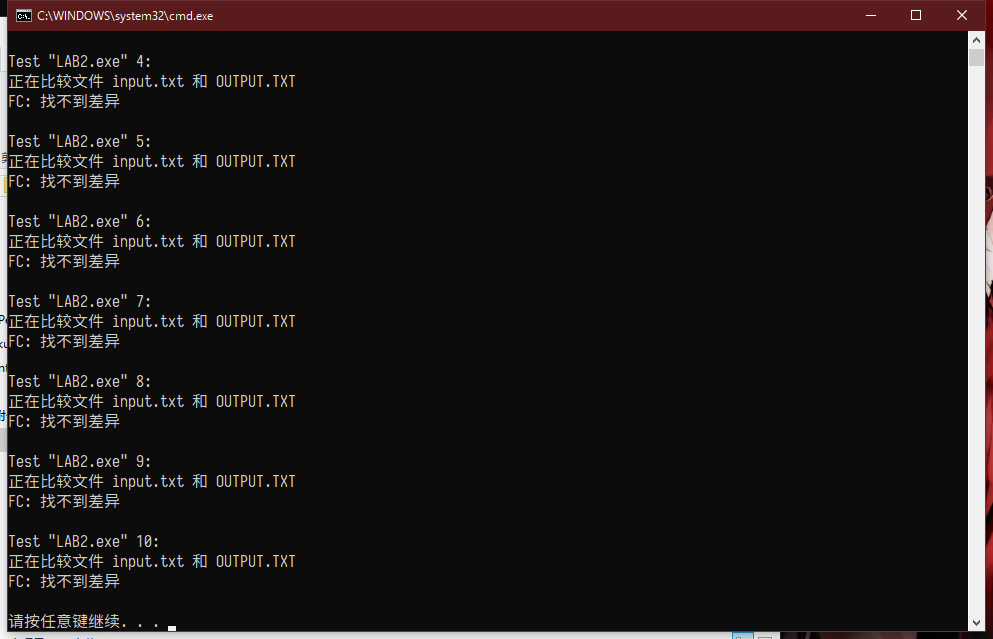


图2-5 check\_win测试SR

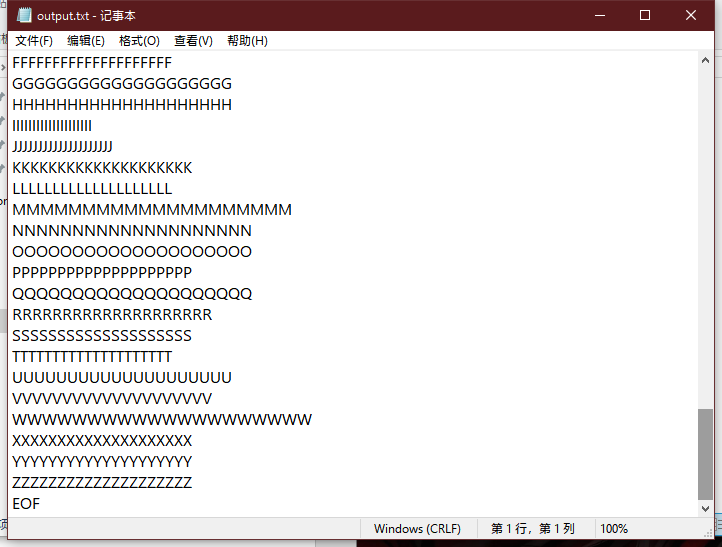


图2-6 测试结果

### **2.3.3 简单TCP/IP协议的设计、验证及结果分析**

简单TCP中，引入快速重传，即接收到3个冗余ACK时，立刻按照该ACK重传报文。

当定时器超时后，重启定时器，重传最早发送且未被确认的报文。

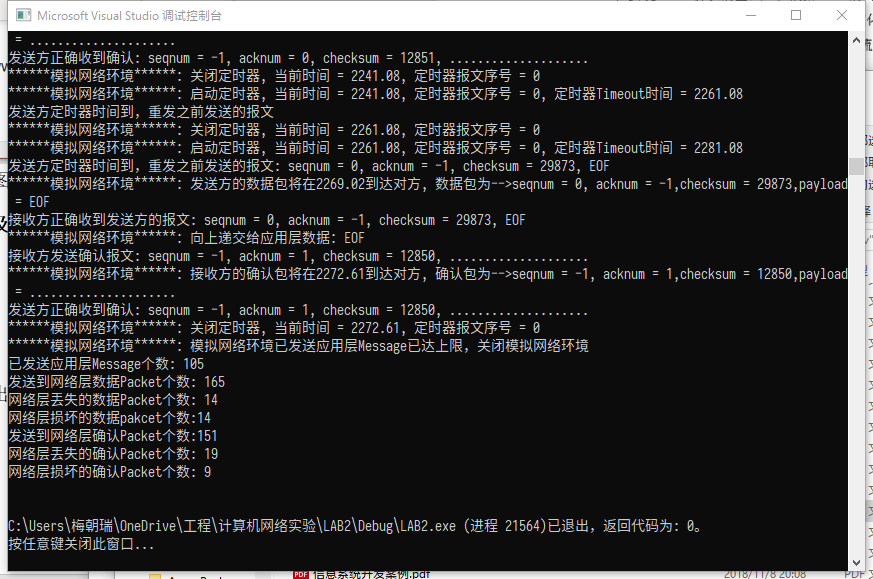


图2-7 TCP单独执行测试

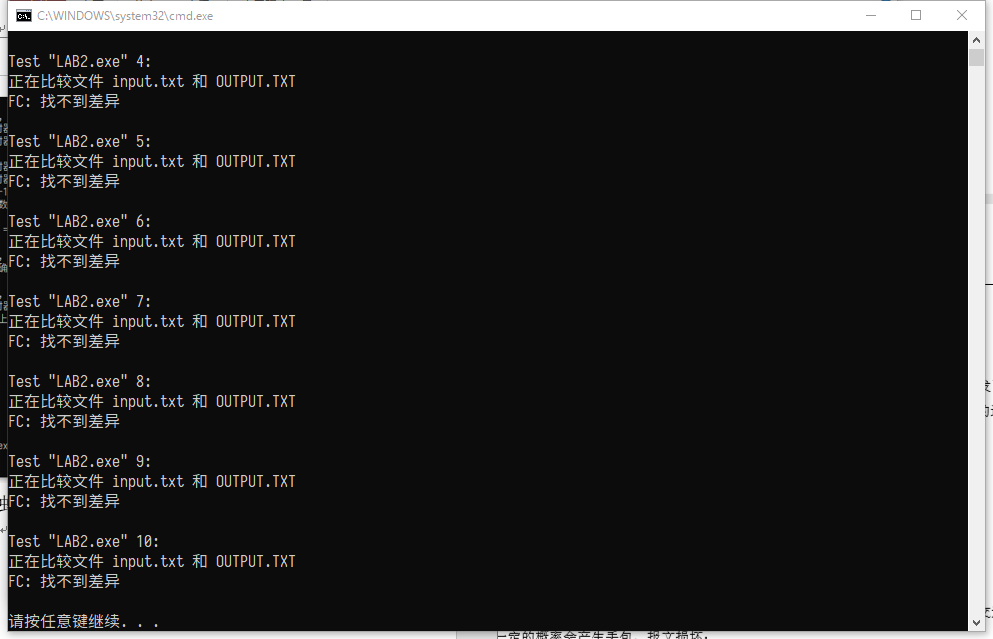


图2-8 check\_win测试TCP

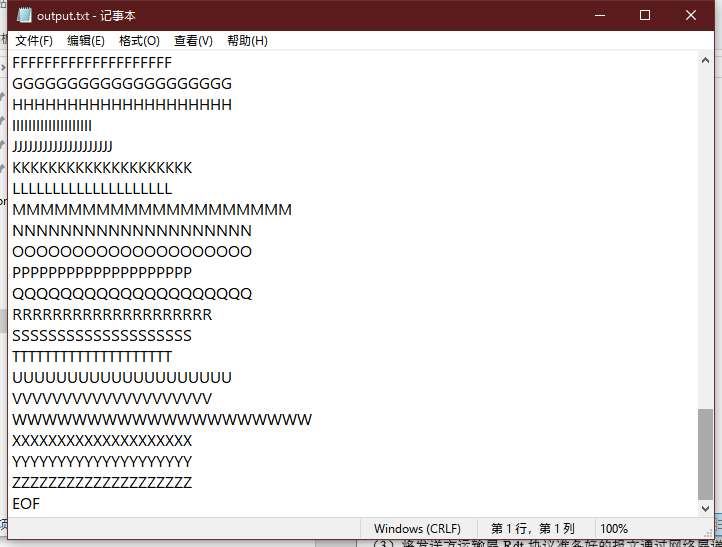


图2-9 测试结果

## **2.4 其它需要说明的问题**

在main函数中，用注释的方式给出了三个协议的选择功能，编译前取消掉其中一个协议前的注释，即按该协议进行编译。

# 实验三 基于CPT的组网实验

## **3.1 环境**

AMD Ryzen 5 1600 Six-Core Processor 3.20 GHz

Windows 10 专业版 Build 17763.195

Cisco Packet Tracer 6.0.0.0045

## **3.2 实验要求**

熟悉 Cisco Packet Tracer 仿真软件。

利用 Cisco Packet Tracer 仿真软件完成实验内容。

**基本部分：**

本部分实验为基础部分的实验，分为两项内容，每项实验内容在最终的评价中占比 30%，本部分实验将使用两张拓扑结构图配合完成实验，如图3-1和3-2所示：

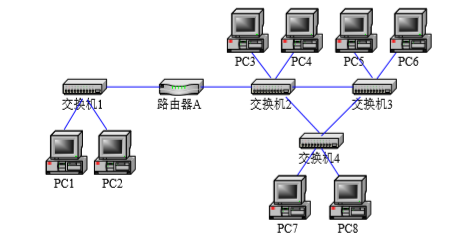


图3-1 参考拓扑1

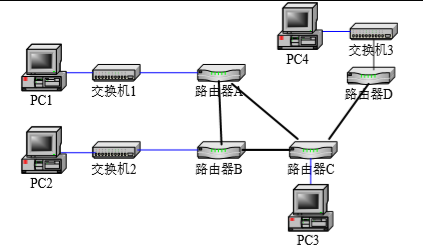


图3-2 参考拓扑2

**第一项实验——IP 地址规划与 VLan 分配实验：**

 使用仿真软件描述网络拓扑图 1.1。

** 基本内容 1**

1. 将 PC1、PC2 设置在同一个网段，子网地址是：192.168.0.0/24;
2. 将 PC3~PC8 设置在同一个网段，子网地址是：192.168.1.0/24;
3. 配置路由器，使得两个子网的各 PC 机之间可以自由通信。

** 基本内容 2**

1. 将 PC1、PC2 设置在同一个网段，子网地址是：192.168.0.0/24；
2. 将 PC3、PC5、PC7 设置在同一个网段，子网地址是：192.168.1.0/24；
3. 将 PC4、PC6、PC8 设置在同一个网段，子网地址是：192.168.2.0/24；
4. 配置交换机 1、2、3、4，使得 PC1、PC2 属于 Vlan2，PC3、PC5、PC7 属于 Vlan3，PC4、PC6、PC8属于 Vlan4；
5. 测试各 PC 之间的连通性，并结合所学理论知识进行分析；
6. 配置路由器，使得拓扑图上的各 PC 机之间可以自由通信，结合所学理论对你的路由器配置过程进行详细说明。

**第二项实验——路由配置实验**

 使用仿真软件描述网络拓扑图 1.2

** 基本内容 1**

1. 将 PC1 设置在 192.168.1.0/24 网段；
2. 将 PC2 设置在 192.168.2.0/24 网段；
3. 将 PC3 设置在 192.168.3.0/24 网段；
4. 将 PC4 设置在 192.168.4.0/24 网段
5. 设置路由器端口的 IP 地址
6. 在路由器上配置 RIP 协议，使各 PC 机能互相访问

** 基本内容 2**

1. 将 PC1 设置在 192.168.1.0/24 网段；
2. 将 PC2 设置在 192.168.2.0/24 网段；
3. 将 PC3 设置在 192.168.3.0/24 网段；
4. 将 PC4 设置在 192.168.4.0/24 网段
5. 设置路由器端口的 IP 地址
6. 在路由器上配置 OSPF 协议，使各 PC 机能互相访问

** 基本内容 3**

1. 在基本内容 1 或者 2 的基础上，对路由器 1 进行访问控制配置，使得 PC1 无法访问其它 PC，也不能被其它 PC 机访问。
2. 在基本内容 1 或者 2 的基础上，对路由器 1 进行访问控制配置，使得 PC1 不能访问 PC2，但能访问其它 PC 机。

**综合部分：**

**实验背景：**

某学校申请了一个前缀为 211.69.4.0/22 的地址块，准备将整个学校连入网络。该学校有 4个学院，1个图书馆，3个学生宿舍。每个学院有 20 台主机，图书馆有 100 台主机，每个学生宿舍拥有 200 台主机。

**组网需求：**

1. 图书馆能够无线上网
2. 学院之间可以相互访问
3. 学生宿舍之间可以相互访问
4. 学院和学生宿舍之间不能相互访问
5. 学院和学生宿舍皆可访问图书馆。

**实验任务要求：**

1. 完成网络拓扑结构的设计并在仿真软件上进行绘制(要求具有足够但最少的设备，不需要考虑设备冗余备份的问题)
2. 根据理论课的内容，对全网的 IP 地址进行合理的分配
3. 在绘制的网络拓扑结构图上对各类设备进行配置，并测试是否满足组网需求，如有无法满足之处，请结合理论给出解释和说明

## **3.3 基本部分实验步骤说明及结果分析**

### **3.3.1 IP地址规划与VLan分配实验的步骤及结果分析**

在Cisco Packet Tracer中按照图3-1搭建出基本的网络拓扑图，如图3-3所示：

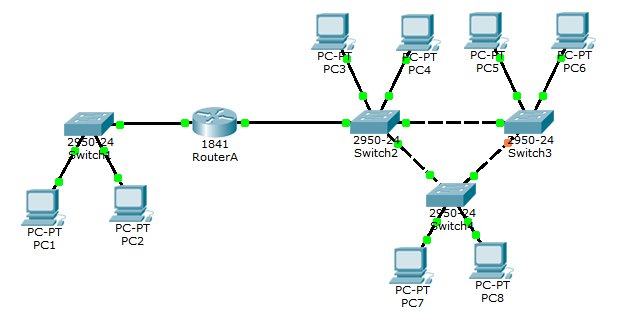


图3-3 IP地址规划实验拓扑图

设定路由器接口IP，左侧子网设为192.168.0.1，右侧子网设为192.168.1.1，均作为子网中其他主机的默认网关。

为两边子网内的主机设定IP，归入对应子网，子网掩码均为255.255.255.0。

至此，子网内部已经可以相互ping通。

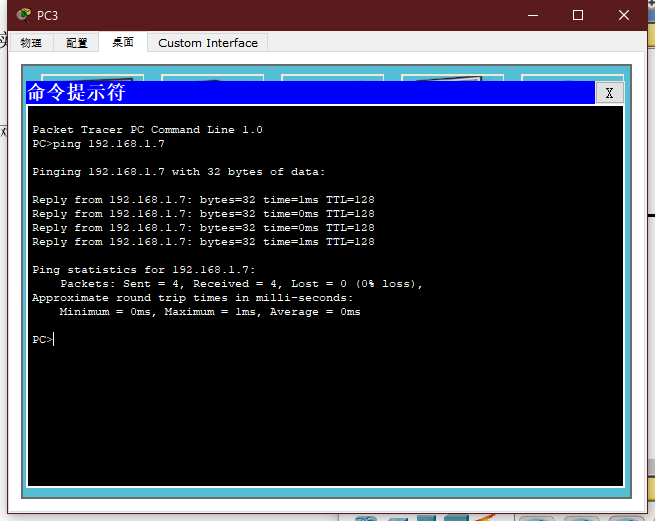


图3-4 实验1子网内部ping

实现基本内容2时，更改右侧PC4,PC6,PC8的IP和默认网关即可。

下面进行VLAN设置。

在switch1上设置vlan2，将PC1和PC2以Access接入vlan2。

在switch2、3、4上设置vlan3、vlan4,将PC3，PC5，PC7以Access接入vlan3，PC4,PC6,PC8以Access接入vlan4。

将switch与switch之间、switch与router之间的所有连接类型设定为Trunk，选择所有VLAN。

为了使两个子网可以通过同一接口，下面需要进行子接口划分。

配置时，在路由器的命令行中输入以下命令：

Router(config)# int fa0/1

Router (config-if)# no shutdown   
Router (config-if)# exit   
Router (config)# int fa0/1.1   
Router (config-subif)# ip addr 192.168.1.1 255.255.255.0   
Router (config-subif)# exit   
Router (config)# int fa0/1.2   
Router (config-subif)# ip addr 192.168.2.1 255.255.255.0   
Router (config-subif)# end

Router (config)# int fa0/1  
Router (config-if)# no shutdown

Router (config-if)# exit   
Router (config)# int fa0/1.1  
Router (config-if)# encap dot1q 3   
Router (config-if)# exit   
Router (config)# int fa0/1.2  
Router (config-if)# encap dot1q 4   
Router (config)# no shutdown

依照上面的方法，在左侧接口处也划分出一个子接口。配置完成后，路由器上显示的信息：

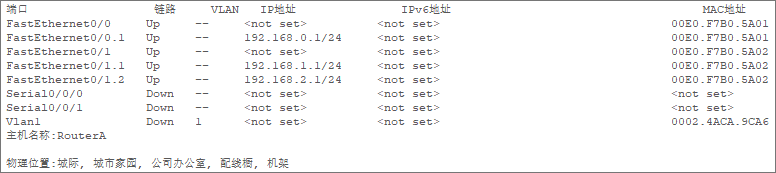


图3-5 路由器配置信息

至此，三个子网之间已经可以互相ping通。用PC2连接PC8：

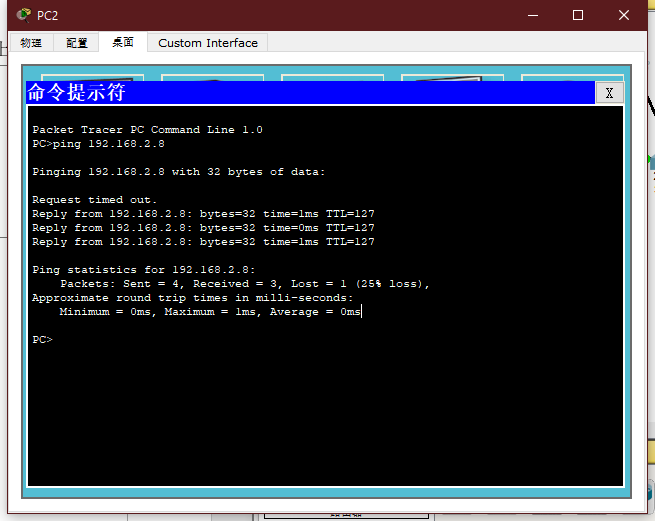


图3-6 子网连通性测试

IP地址规划与VLAN分配实验完成。

### **路由配置实验的步骤及结果分析**

首先按照图3-2搭建基本网络拓扑图，如图3-7所示：

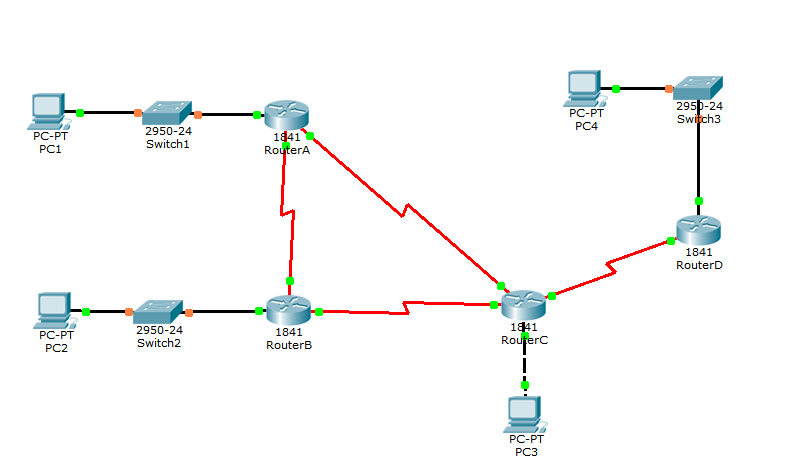


图3-7 路由配置实验拓扑图

为PC1、PC2、PC3、PC4设置IP地址和默认网关。

在路由器上设置接口IP作为默认网关。

在路由器上设置RIP协议（录入相连的子网网络号）。

完成后，至此各主机已可以连通。测试以PC2连接PC4：

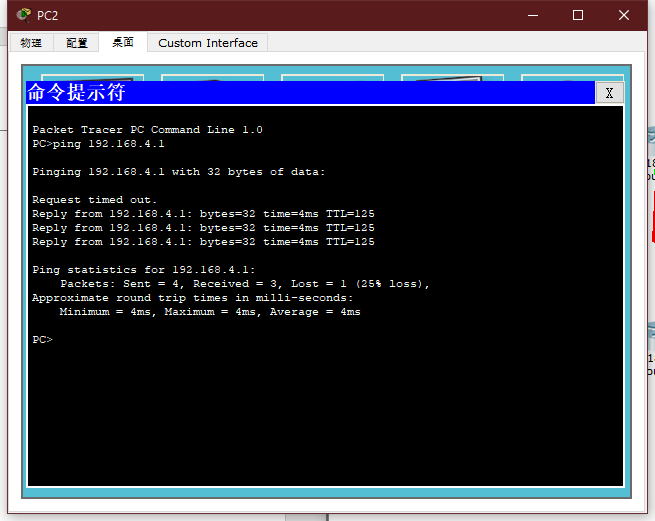


图3-8 连通性测试

若使用OSPF路由协议，则需要使用命令行模式设置。

例如，对RouterA，进行如下的命令配置：

Router#conf t //进入全局配置模式

Router(config)#router ospf 1 //选择ospf协议

Router(config-router)#network 192.168.1.0 0.0.0.255 area 0

Router(config-router)#network 192.168.5.0 0.0.0.255 area 0

Router(config-router)#network 192.168.7.0 0.0.0.255 area 0

Router(config-router)#end

Router#copy run startup

在上述命令中，3-5行是为了将与路由器RouterA相连的各个子网加入OSPF协议中。

检测连通性，可确认连通成功。

下面进行访问控制。

对路由器Router A与PC1相连的端口fa0/0进行配置，使得fa0/0内的主机无法访问其他的网段。对RouterA进行命令行配置，配置命令如下：

Router#conf t

Router(config)#access-list 10 deny 192.168.1.0 0.0.0.255

Router1(config-if)#access-list 10 permit any

Router1(config-if)#interface fa0/0

Router1(config-if)#ip access-group 10 in

使用ACL进行访问控制配置，让PC1也不同能其他主机访问。对RouterA进行命令行配置，限制fa0/0端口处的out。配置命令如下：

Router#conf t

Router(config)#access-list 101 deny ip any 192.168.1.0 0.0.0.255

Router(config)#access-list 101 permit ip any any

Router1(config-if)#interface fa0/0

Router1(config-if)#ip access-group 101 out

至此，PC1已不能与其他主机通讯，如图3-9所示：

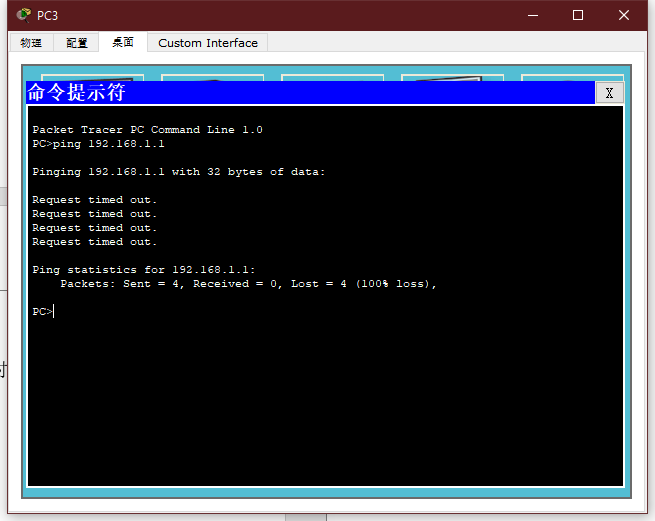


图3-9 ACL限制PC1通讯

要使PC1不能访问PC2，只需设置对PC2的限制即可：

Router#conf t

Router(config)#access-list 101 deny 192.168.1.1 0.0.0.255 192.168.2.1 0.0.0.255

Router1(config-if)#access-list 101 permit any

Router1(config-if)#interface fa0/0

Router1(config-if)#ip access-group 10 in

测试结果如图3-10：

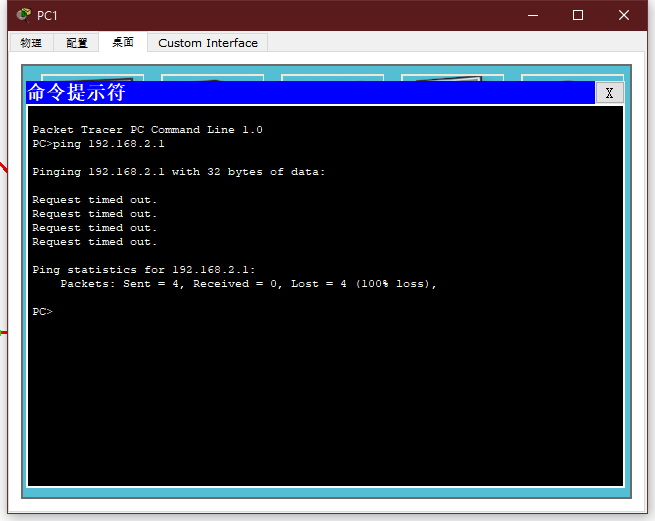


图3-10 ACL限制PC1与PC2连接

至此，路由配置实验结束。

## **综合部分实验设计、实验步骤及结果分析**

### **实验设计**

为需求量为200的部门分配/25的网段，为需求量为100的部门分配/27的网段，即可满足需求。

另外，由于要求图书馆能支持无线上网，需要在图书馆子网中设置无线接入点和DHCP服务器。

设计基本拓扑图如下：

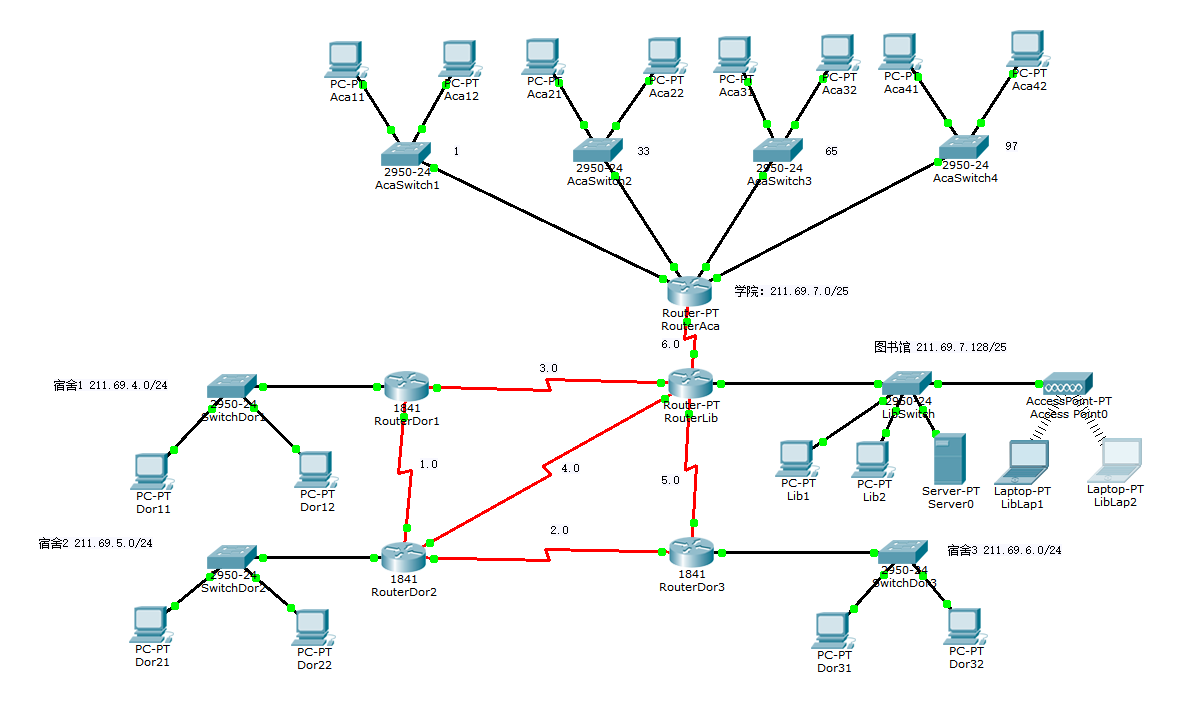


图3-11 综合实验拓扑图

### **实验步骤**

依照设计作出拓扑图。

对路由器端口进行IP配置，路由器间使用OSPF选路算法。

对四个学院使用4个VLAN，三个宿舍使用3个VLAN，图书馆另设一个VLAN，便于进行访问控制。

设置宿舍和学院网关路由器的ACL，使宿舍与学院不能互相访问。

配置图书馆的DHCP服务器和无线接入点（无加密），用搭载无线网卡的Laptop连接无线接入点。

### **结果分析**

测试宿舍区、学院区内部的连通性：

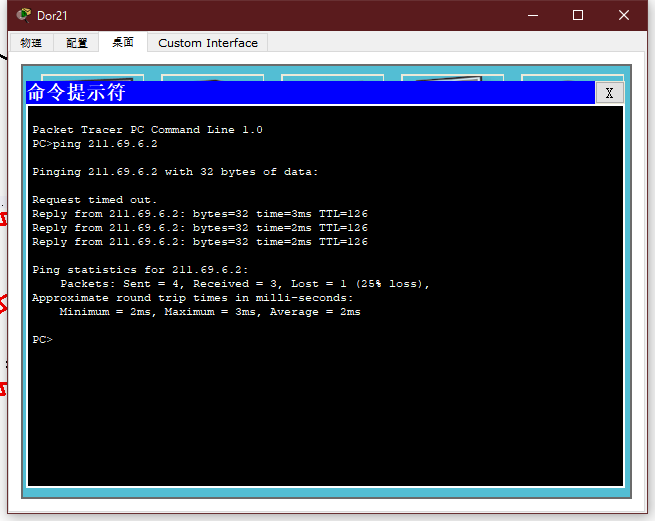


图3-12 宿舍内部连通性测试

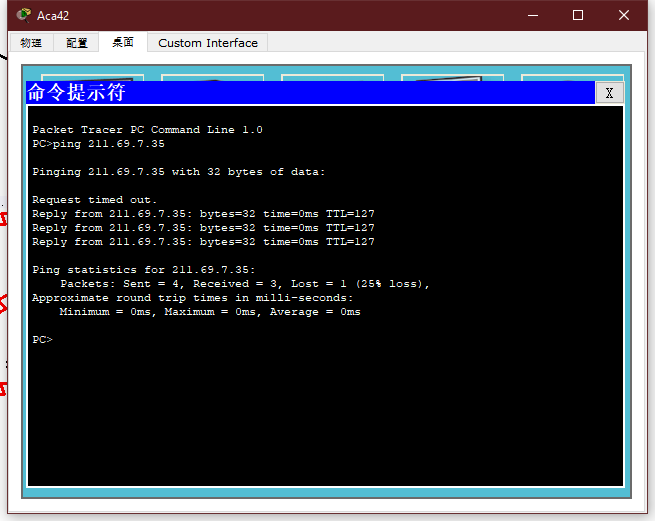


图3-13 学院内部连通性测试

可以看到，宿舍区和学院区内部是可以相互连通的。

接下来尝试从学院区连接宿舍区：

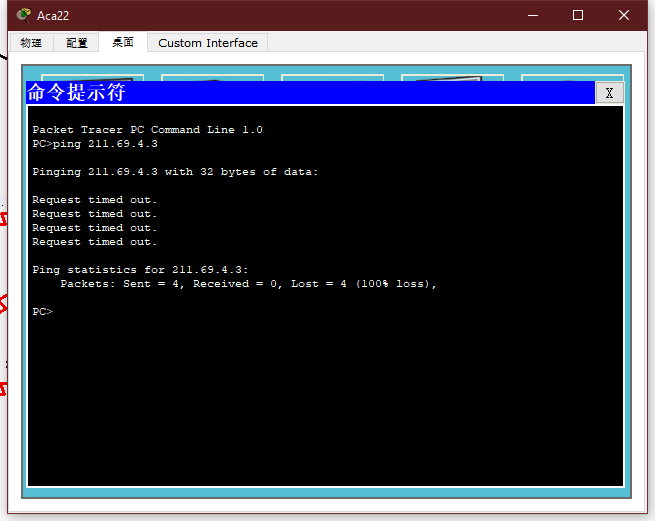


图3-14 测试宿舍与学院连接

满足了宿舍区与学院区无法互联的要求。最后测试用图书馆无线接入的笔记本连接学院区：

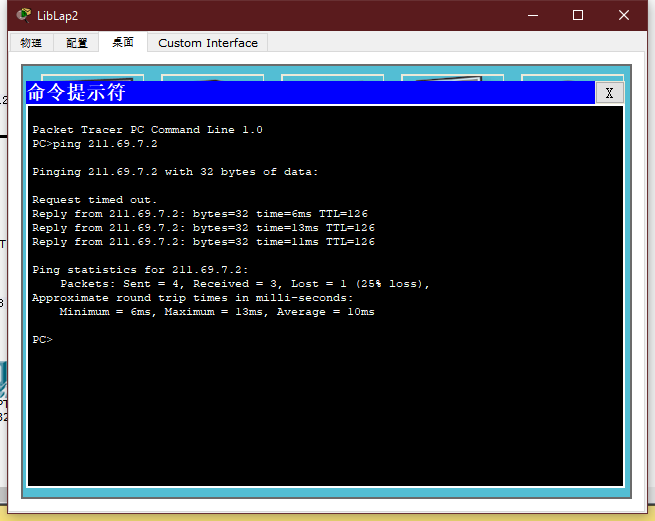


图3-15 图书馆连接测试

连接成功，满足要求，至此综合实验结束。

## **3.5 其它需要说明的问题**

无

# 心得体会与建议

## **4.1 心得体会**

第一次和第三次实验内容都相当有意思，另一方面也相当实用。web服务器实验用基本的库构建了看似复杂的web服务器，让我体会到服务器的本质就是根据客户端请求提供相应服务（大多数情况下是传输某些文件）。组网综合实验简直将各种零散的网络知识都融会贯通到了一起，完成实验的成就感和知识的条理整合感都是很大的。

第二次实验虽然看起来非常硬核，好在后来发现提供了停等协议的范例，由此魔改出回退N步，选择重传和TCP就都简单了起来，最重要的是理解这几种协议的思想和特色功能，以及它们是为了解决什么问题而设定出来的。

## **4.2 建议**

第三次实验是我遇到问题最大的一次，主要原因是关于VLAN、子接口和连接类型文档中在哪里都没有给出说明，走了很多弯路，直到读了前几年的学长的一份实验报告才知道了那些神秘的指令，解决了问题。

希望以后能把参考指令表的那部分写得更加详细，或者给出Cisco官网上的官方文档地址以便于查阅。