

计算机网络编程

第14章 FTP客户机程序设计

信息工程学院 方徽星

fanghuixing@hotmail.com

大纲

- 设计目的
- 相关知识
- 例题分析

1. 设计目的

- **通过FTP客户机程序的设计**
 - **了解FTP服务的基本概念与主要功能**
 - **掌握应用层服务的设计思路与编程方法**

2. 相关知识： 应用层的基本概念

- **应用层**

- **是网络体系结构的最高层次**
- **提供各种类型的网络服务**
- **每种应用层服务都有对应的协议标准**

2. 相关知识：应用层的基本概念

• 应用层协议

- 文件传输协议 (FTP)
 - 远程登录 (Telnet)
 - 简单邮件传输协议 (SMTP)
 - 超文本传输协议 (HTTP)
 - 简单网络管理协议 (SNMP)
 - 简单文件传输协议 (TFTP)
 - 域名系统 (DNS)
-
- 只依赖TCP
- 只依赖UDP
- TCP/UDP

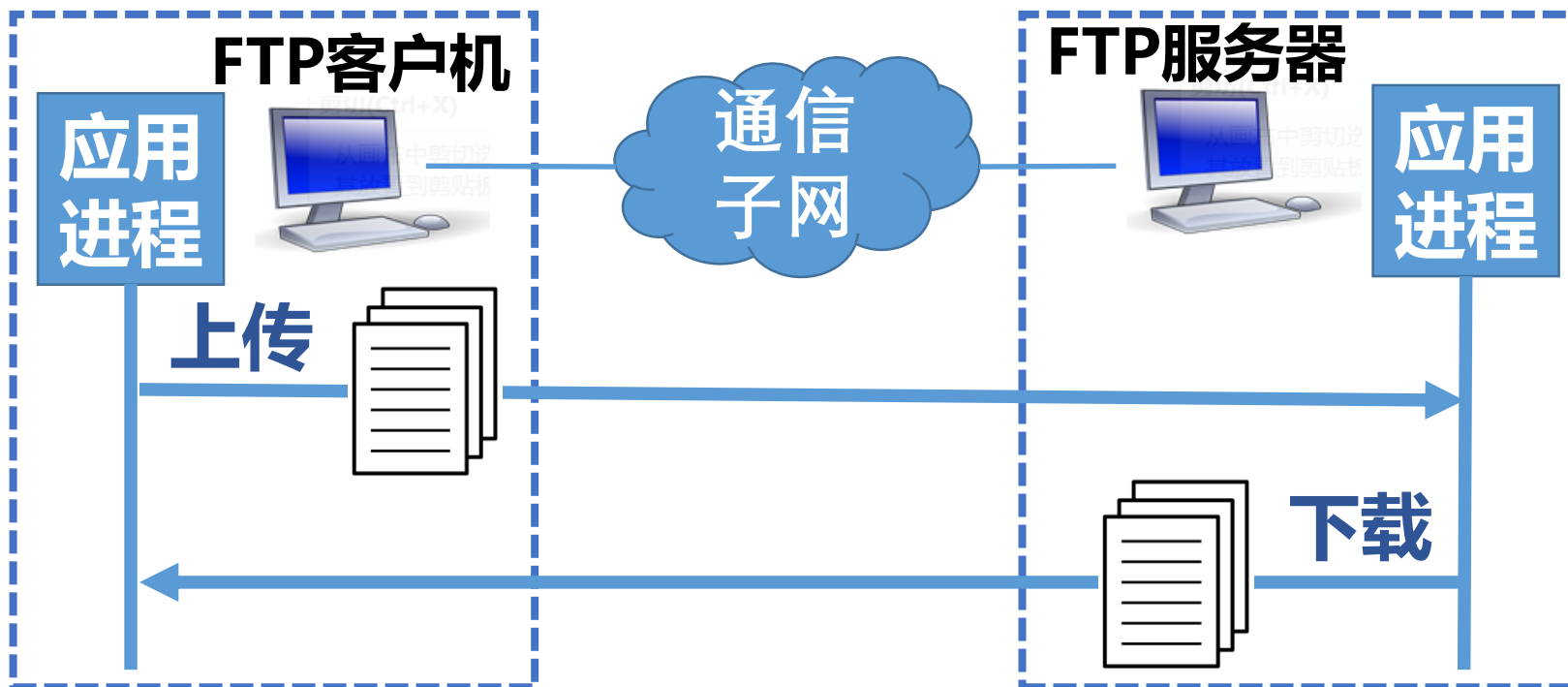
2. 相关知识： FTP服务的基本概念

- **FTP服务： 文件传输服务**
 - **遵循FTP协议 (File Transfer Protocol)**
 - **允许用户将文件从一台计算机传输到另一台计算机，
保证传输的可靠性**

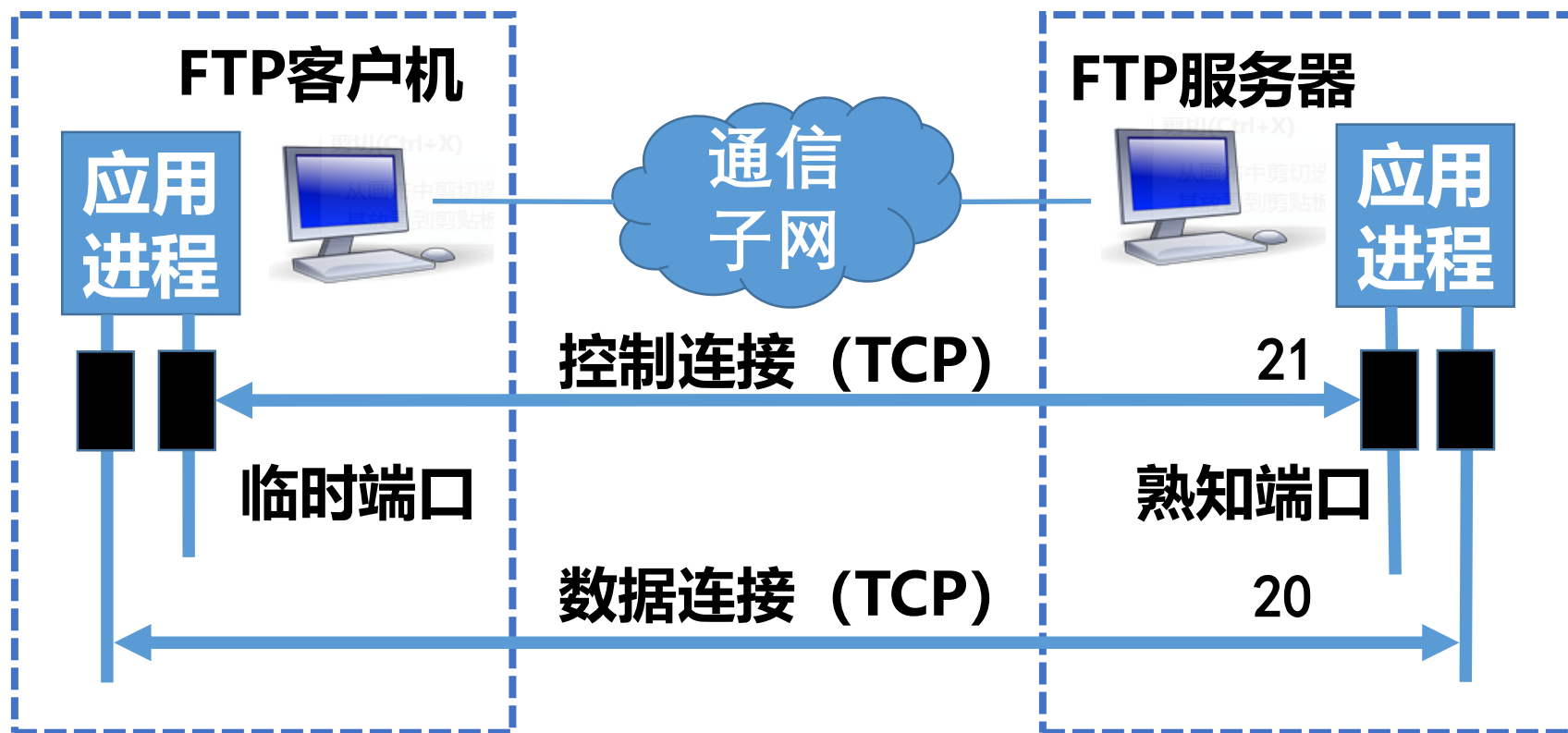
2. 相关知识：FTP服务的基本概念

- **FTP服务：文件传输服务**

- 下载：文件从FTP服务器传输到FTP客户机
- 上传：文件从FTP客户机传输到FTP服务器

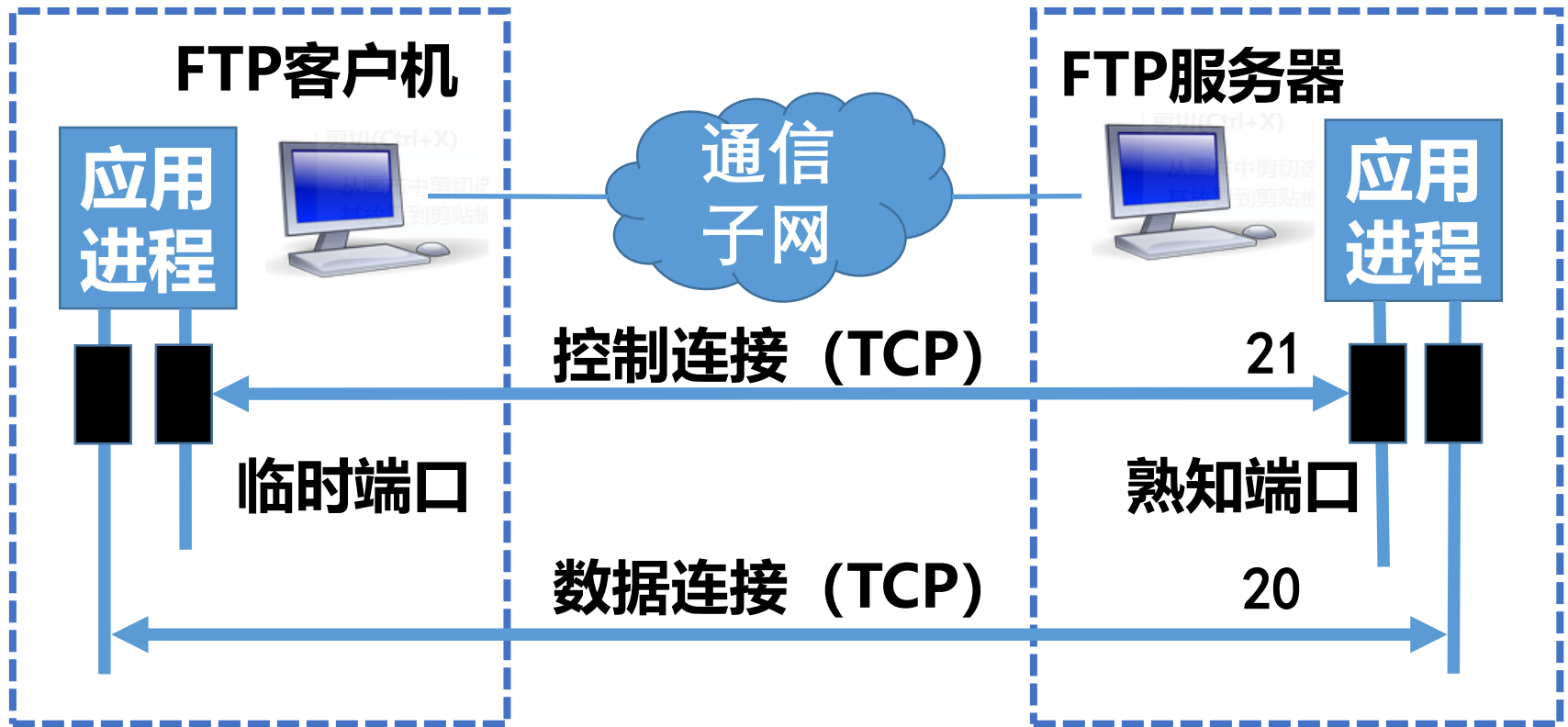


2. 相关知识：FTP服务的工作原理



控制连接：传输FTP命令与应答信息、完成连接建立、身份认证、异常处理等控制操作

2. 相关知识：FTP服务的工作原理



数据连接：传输文件或目录信息

2. 相关知识：FTP服务的工作原理

- FTP协议规定了控制连接和数据连接建立和释放的顺序

规则1

先建立控制连接，然后才建立数据连接

规则2

先释放数据连接，然后再释放控制连接

2. 相关知识：FTP服务的工作原理

- FTP协议规定了控制连接和数据连接建立和释放的顺序

规则3

建立数据连接后，才能传输数据，且数据传输过程中，须保持控制连接不中断

2. 相关知识：FTP服务的工作原理

- 连接/释放的发起者规则

规则1

控制连接与数据连接建立的发起者
只能是FTP客户机

规则2

控制连接释放的发起者
只能是FTP客户机

2. 相关知识：FTP服务的工作原理

- 连接/释放的发起者规则

规则3

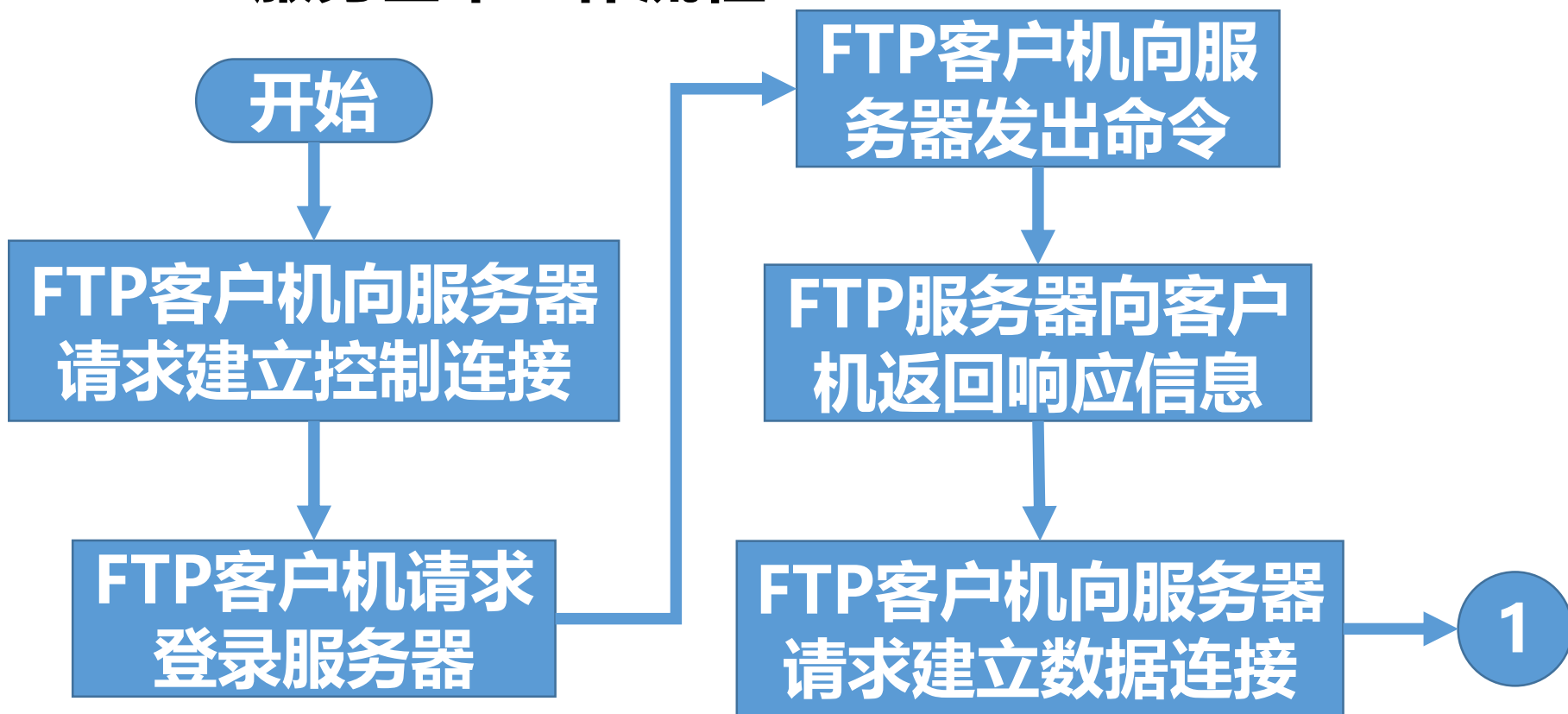
数据连接释放的发起者可以是
FTP客户机 或 FTP服务器

规则4

在数据连接保持的情况下，控制连接中断，
则FTP服务器可以要求释放数据连接

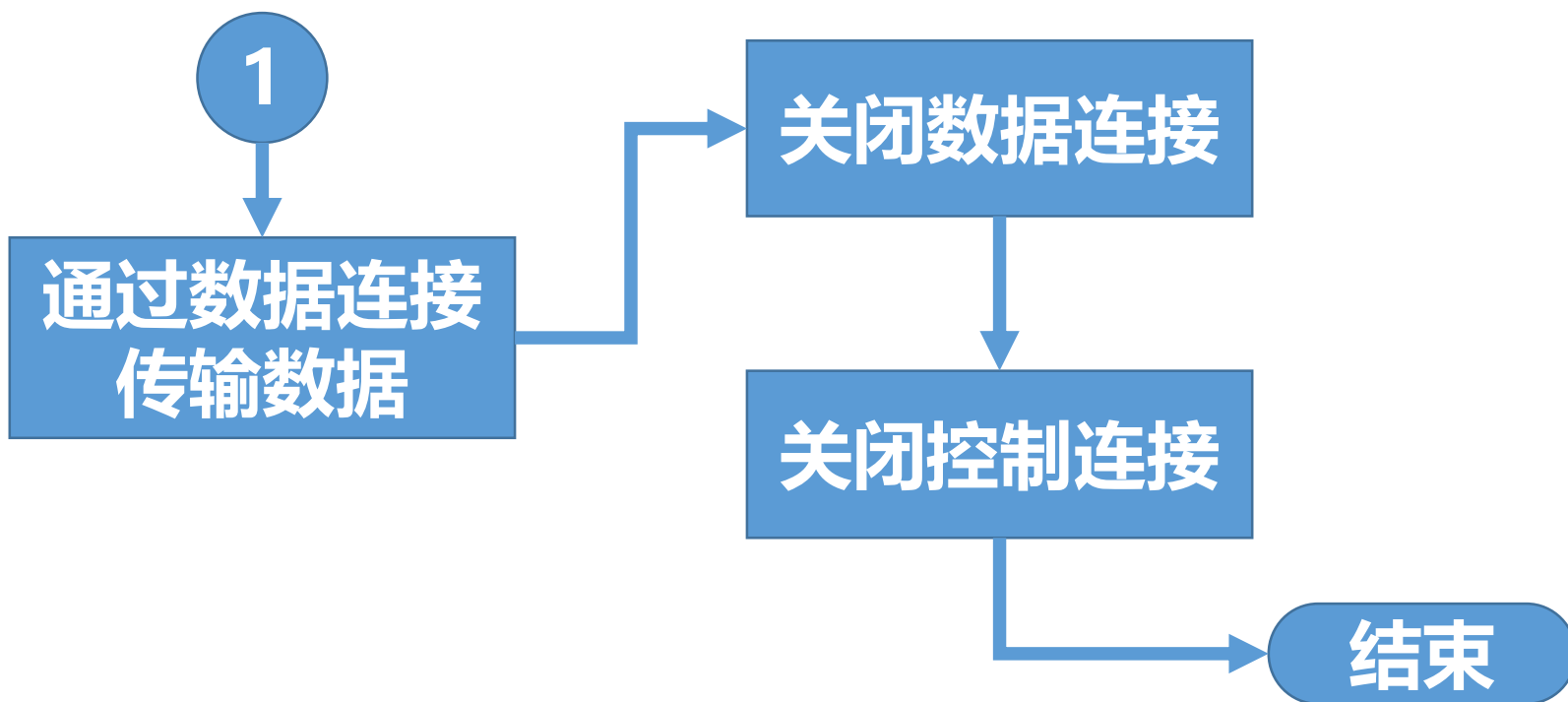
2. 相关知识：FTP服务的工作原理

• FTP服务基本工作流程



2. 相关知识：FTP服务的工作原理

- FTP服务基本工作流程



2. 相关知识：FTP命令与应答

- FTP命令

命令名 <参数>

Access commands

<i>Command</i>	<i>Argument(s)</i>	<i>Description</i>
USER	User id	User information
PASS	User password	Password
ACCT	Account to be charged	Account information
REIN		Reinitialize
QUIT		Log out of the system
ABOR		Abort the previous command

<https://tools.ietf.org/html/rfc959>

File management commands

<i>Command</i>	<i>Argument(s)</i>	<i>Description</i>
CWD	Directory name	Change to another directory
CDUP		Change to parent directory
DELE	File name	Delete a file
LIST	Directory name	List subdirectories or files
NLIST	Directory name	List subdirectories or files without attributes
MKD	Directory name	Create a new directory
PWD		Display name of current directory
RMD	Directory name	Delete a directory
RNFR	File name (old)	Identify a file to be renamed
RNTO	File name (new)	Rename the file
SMNT	File system name	Mount a file system

Data formatting commands

<i>Command</i>	<i>Argument(s)</i>	<i>Description</i>
TYPE	A (ASCII), E (EBCDIC), I (Image), N (Nonprint), or T (TELNET)	Define file type
STRU	F (File), R (Record), or P (Page)	Define organization of data
MODE	S (Stream), B (Block), or C (Compressed)	Define transmission mode

Port defining commands

<i>Command</i>	<i>Argument(s)</i>	<i>Description</i>
PORT	6-digit identifier	Client chooses a port
PASV		Server chooses a port

File transfer commands

<i>Command</i>	<i>Argument(s)</i>	<i>Description</i>
RETR	File name(s)	Retrieve files; file(s) are transferred from server to client
STOR	File name(s)	Store files; file(s) are transferred from client to server
APPE	File name(s)	Similar to STOR, but if file exists, data must be appended to it
STOU	File name(s)	Same as STOR, but file name will be unique in the directory
ALLO	File name(s)	Allocate storage space for files at the server
REST	File name(s)	Position file marker at a specified data point
STAT	File name(s)	Return status of files

2. 相关知识： FTP命令与应答

- FTP应答



<https://tools.ietf.org/html/rfc640>

应答码	描述
Positive Preliminary Reply	
120	Service will be ready shortly
125	Data connection open; data transfer will start shortly
150	File status is OK; data connection will be open shortly

应答码	描述
Positive Completion Reply	
200	Command OK
211	System status or help reply
212	Directory status
213	File status
214	Help message

应答码	描述
Positive Completion Reply	
215	Naming the system type (operating system)
220	Service ready
221	Service closing
225	Data connection open
226	Closing data connection

应答码	描述
Positive Completion Reply	
227	Entering passive mode; server sends its IP address and port number
230	User login OK
250	Request file action OK

应答码	描述
Positive Intermediate Reply	
331	User name OK; password is needed
332	Need account for logging
350	The file action is pending; more information needed

应答码	描述
Transient Negative Completion Reply	
425	Cannot open data connection
426	Connection closed; transfer aborted
450	File action not taken; file not available
451	Action aborted; local error
452	Action aborted; insufficient storage

应答码	描述
Permanent Negative Completion Reply	
500	Syntax error; unrecognized command
501	Syntax error in parameters or arguments
502	Command not implemented
503	Bad sequence of commands
504	Command parameter not implemented

应答码	描述
Permanent Negative Completion Reply	
530	User not logged in
532	Need account for storing file
550	Action is not done; file unavailable
552	Requested action aborted; exceeded storage allocation
553	Requested action not taken; file name not allowed

3. 例题分析：设计要求

- 根据客户机/服务器工作模式
 - 编写FTP客户机程序向服务器发送命令
 - 将FTP服务器返回的应答信息与数据显示在控制台上
- 为简便起见，只需实现USER、PASS、LIST与QUIT命令

3. 例题分析：设计要求

- 具体要求
 - 要求程序为命令程序

FtpClient server_addr



FTP服务器IP地址

3. 例题分析：设计要求

- 具体要求

- 要求将FTP服务器的状态显示在控制台上

```
FTP>Control Connection...
```

```
响应信息...
```

```
FTP>USER:xxxxxx
```

```
响应信息...
```

```
FTP>PASS:xxxxxx
```

```
响应信息...
```

```
FTP>LIST
```

```
响应信息...
```

```
FTP>QUIT
```

```
响应信息...
```

3. 例题分析：关键问题

- 建立控制连接

```
//创建流式套接字
```

```
SocketControl = socket(AF_INET, SOCK_STREAM, 0);
```

```
//填充服务器Socket地址
```

```
sockaddr_in serveraddr;
```

```
serveraddr.sin_family = AF_INET;
```

```
serveraddr.sin_port = htons(21);
```

```
serveraddr.sin_addr.S_un.S_addr = inet_addr(argv[1]);
```


3. 例题分析：关键问题

```
//向FTP服务器发送连接请求
connect( SocketControl,
        (sockaddr*)&serveraddr,
        sizeof(serveraddr));
//从FTP服务器获得连接应答
if(RecvReply())
{
    //判断应答码
    if(nReplyCode==220) //service ready
        ...
    else
        closesocket(SocketControl);
}
```

3. 例题分析：关键问题

- 登录到FTP服务器

```
//构造标准格式的USER命令  
memcpy(Command, "USER " , strlen( "USER " ));  
memcpy(Command + strlen( "USER " ),  
        CmdBuf,  
        strlen(CmdBuf)); //CmdBuf存储用户名  
memcpy(Command  
        +strlen( "USER " )  
        +strlen(CmdBuf),  
        "\r\n" , 2);  
//向FTP服务器发送USER命令  
SendCommand();
```

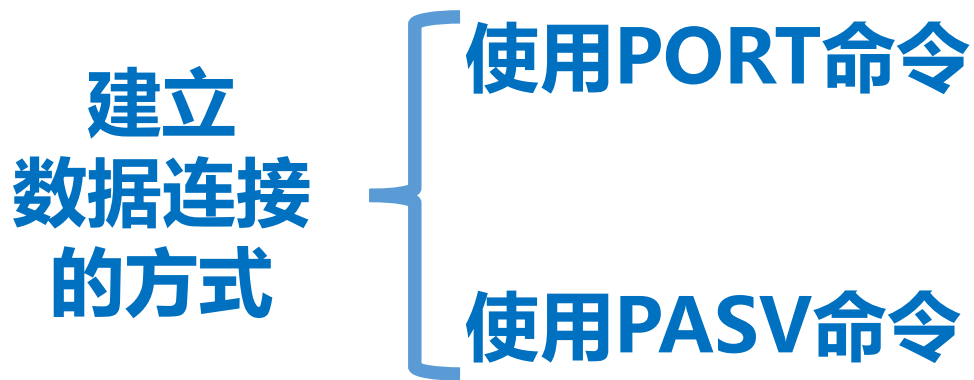
```
//从FTP服务器接收应答信息
if(RecvReply())
{
    //如果登录成功或需要密码
    if(nReplyCode == 230 || nReplyCode == 331 )
        //输出提示信息
}

if (nReplyCode==331)
{
    //发送密码
    ...
}
```

3. 例题分析：关键问题

- 执行LIST命令

- LSIT命令用来返回当前目录中的信息（包括子目录与文件），需要使用数据连接来传输目录信息



3. 例题分析：关键问题

建立数据连接的方式

- 使用PORT命令，**主动模式**，
 - 客户机指定自己用于数据连接的端口
 - 由FTP服务器来与客户机建立数据连接

- 使用PASV命令，**被动模式**，
 - FTP服务器在应答信息中指出用于数据连接的端口
 - 由FTP客户机与服务器建立数据连接

3. 例题分析：关键问题

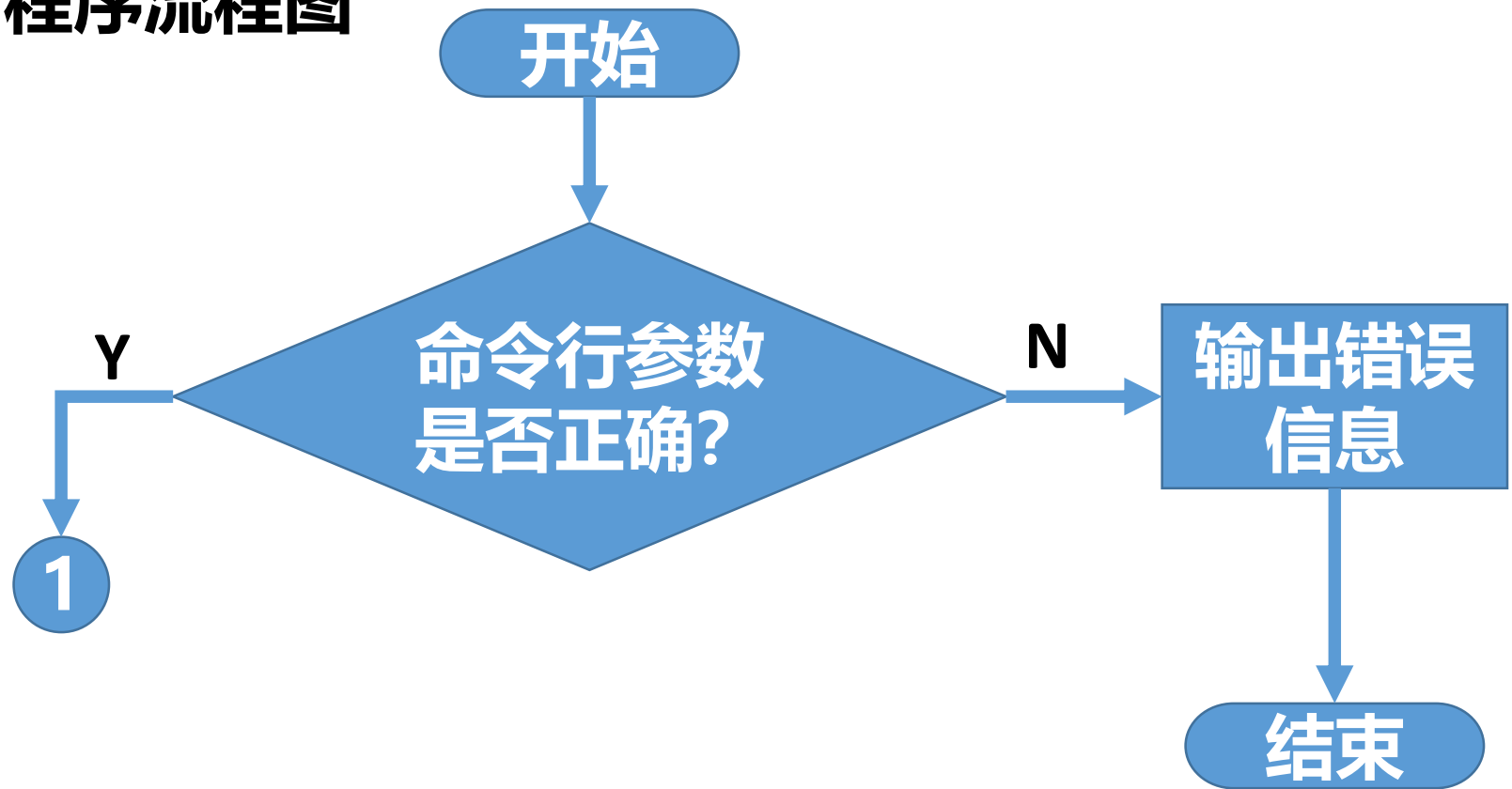
- 执行LIST命令

```
//通过PASV命令获得服务器端口  
SendCommand();  
RecvReply();  
//获取FTP服务器数据端口  
...  
//创建数据连接Socket  
SocketData = socket(AF_INET, SOCK_STREAM, 0);  
//向FTP服务器发送连接请求  
connet(SocketData,  
        (sockaddr*)&serveraddr, sizeof(serveraddr));
```

```
//向FTP服务器发送LIST命令
SendCommand();
//从FTP服务器接收应答信息
if(RecvReply())
{
    if(nReplyCode == 125 || nReplyCode == 150
        || nReplyCode == 226)
        //输出提示信息
}
//获得LIST命令的目录信息
while(true)
{
    memset(ListBuf, 0, MAX_SIZE);
    nRecv = recv(SocketData, ListBuf, MAX_SIZE, 0);
    ...
}
```

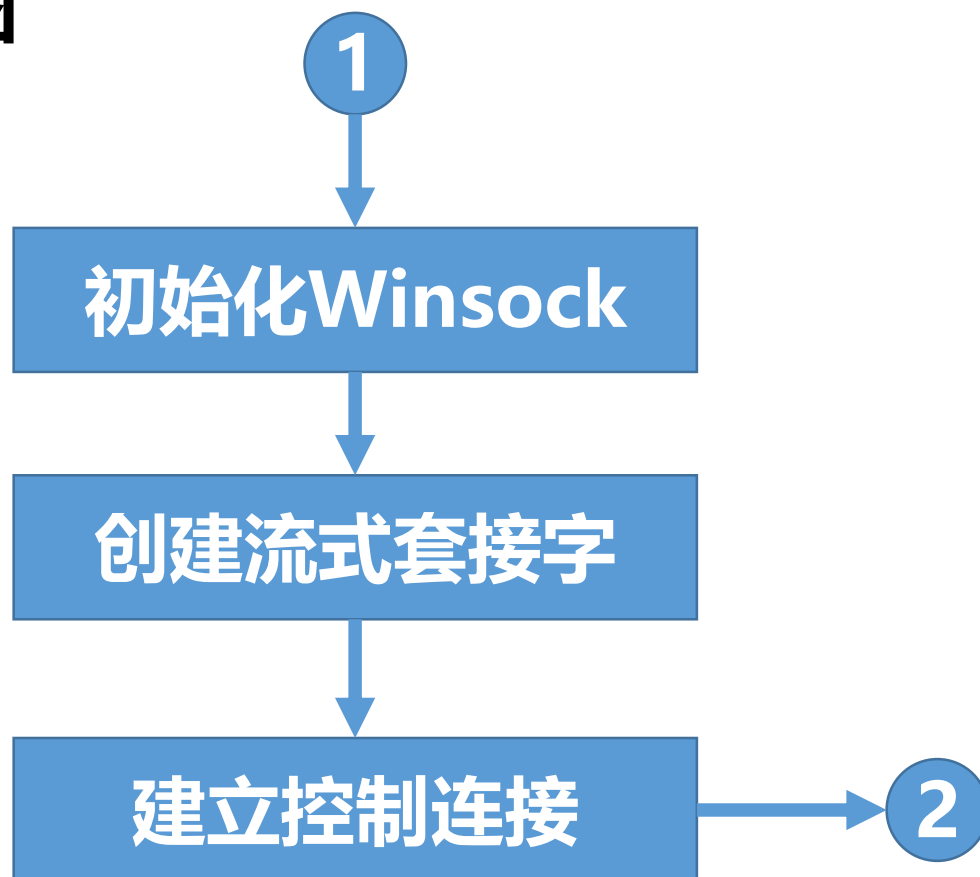
3. 例题分析：关键问题

- 程序流程图



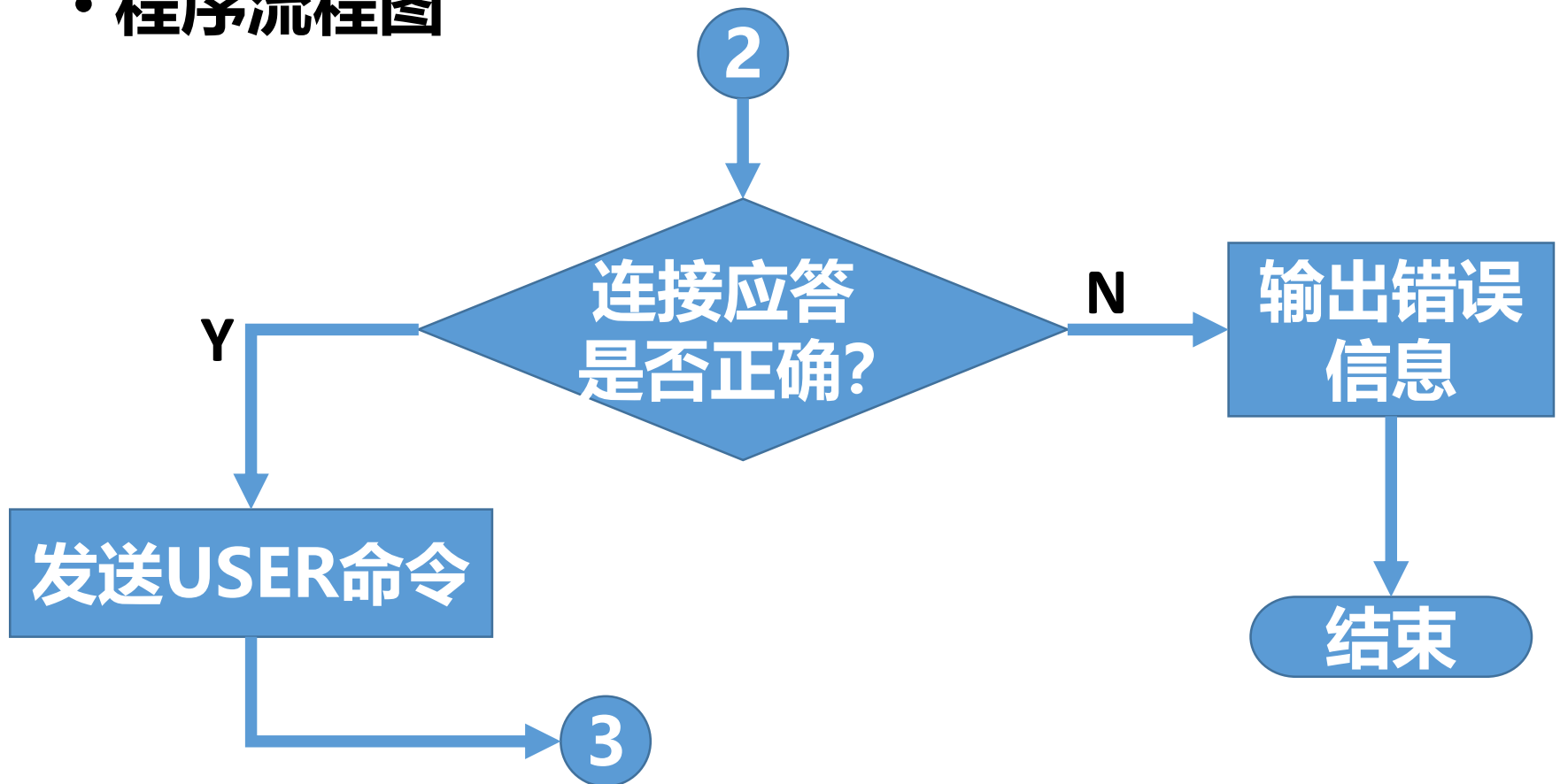
3. 例题分析：关键问题

- 程序流程图



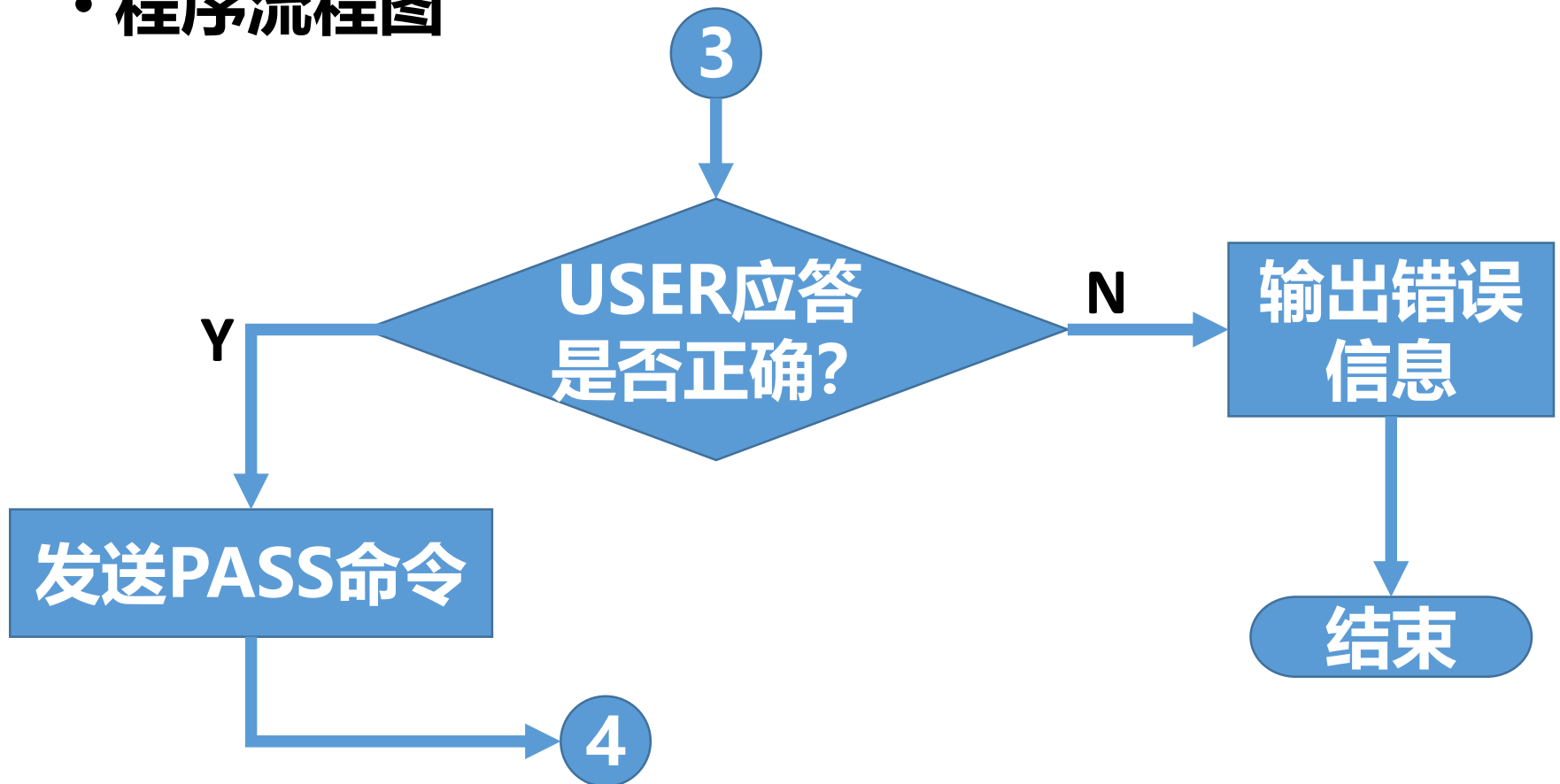
3. 例题分析：关键问题

- 程序流程图



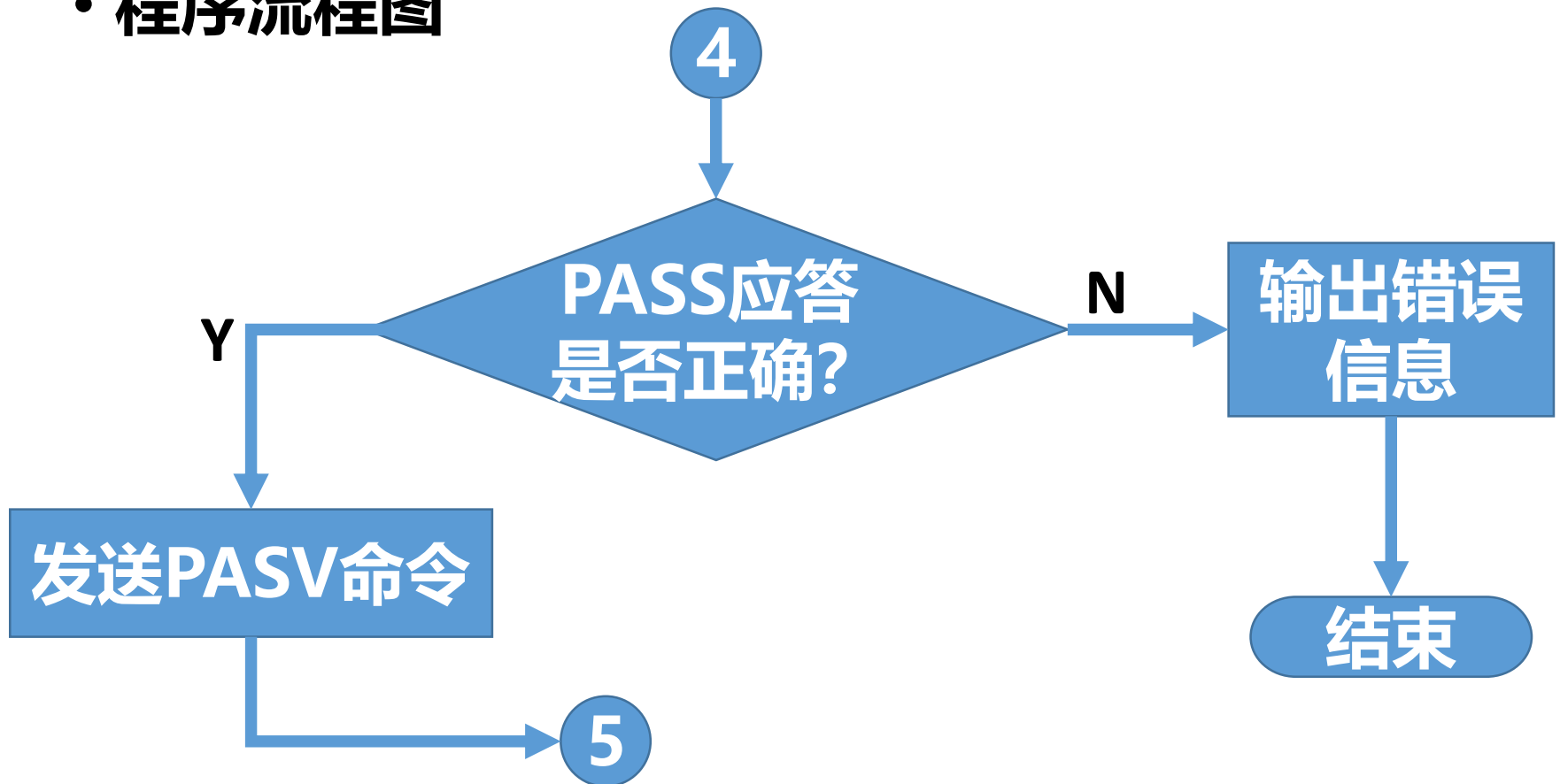
3. 例题分析：关键问题

- 程序流程图



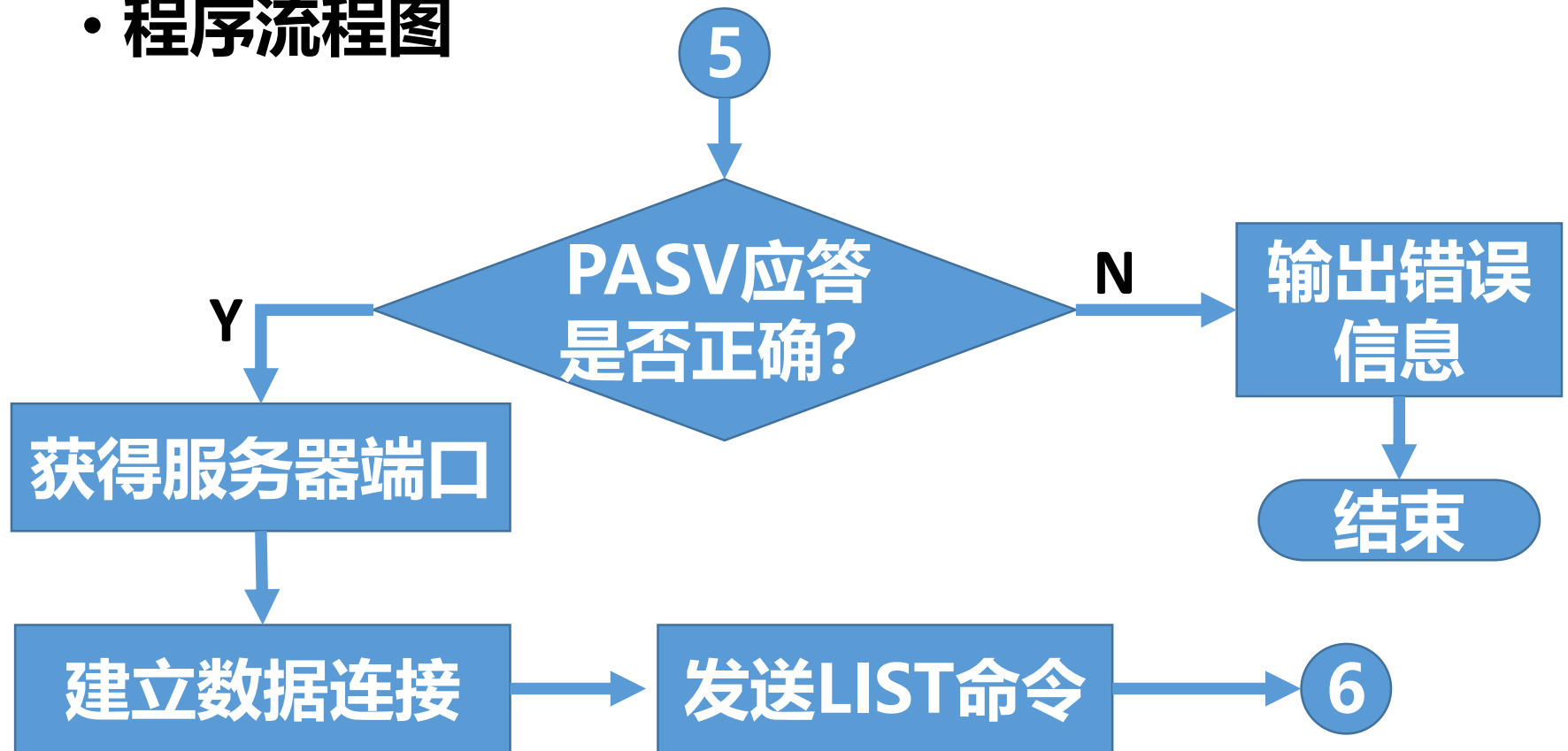
3. 例题分析：关键问题

- 程序流程图



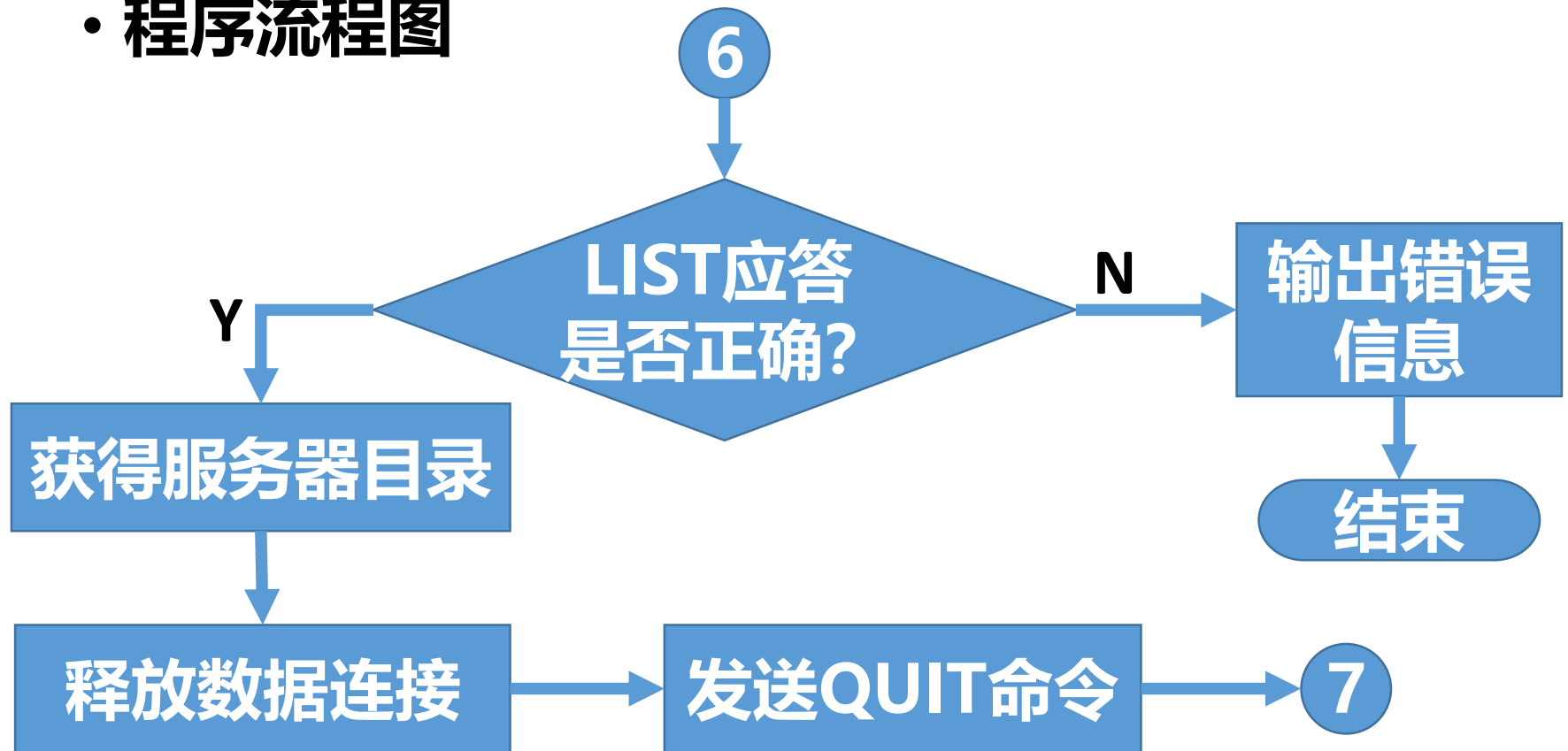
3. 例题分析：关键问题

- 程序流程图



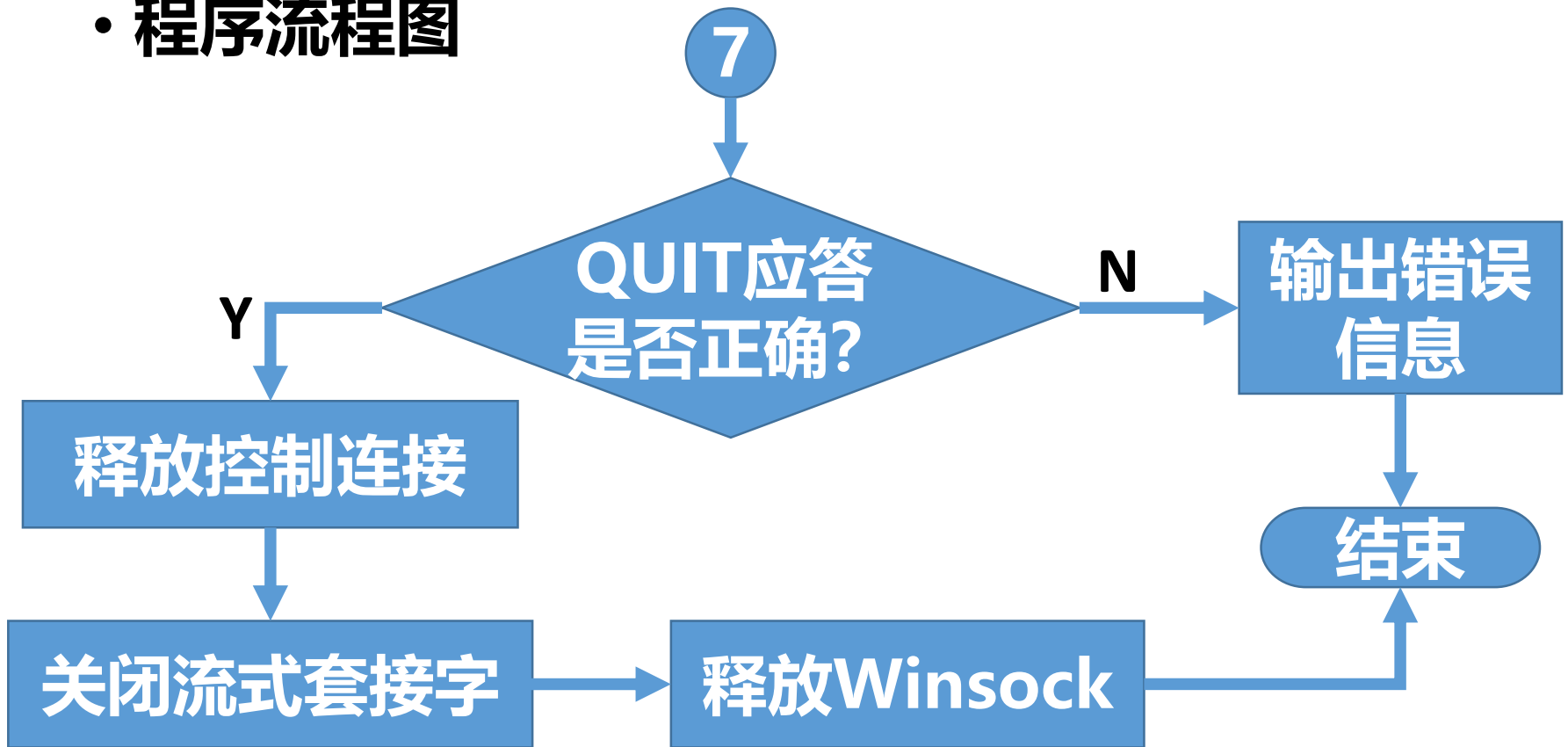
3. 例题分析：关键问题

- 程序流程图

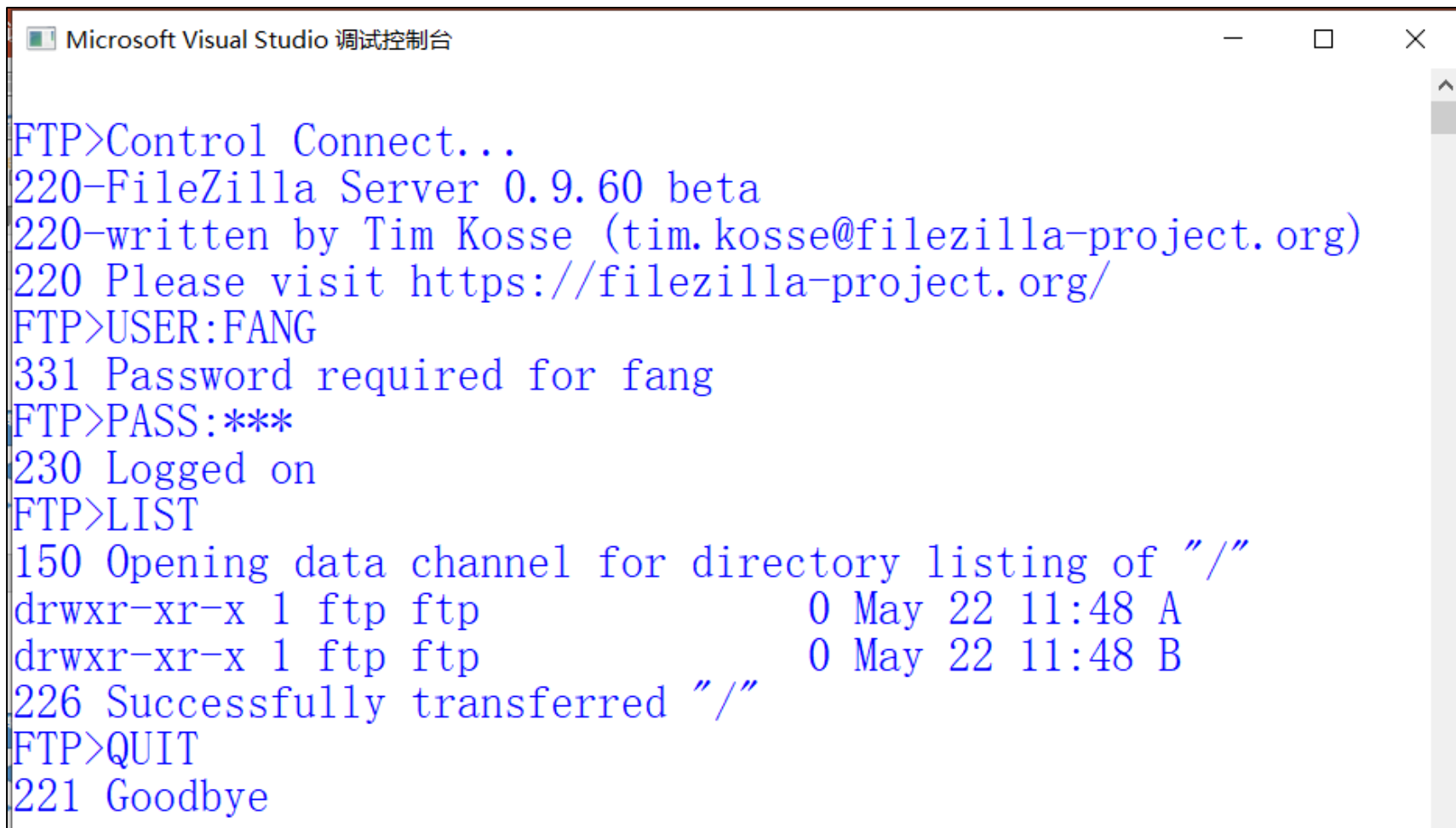


3. 例题分析：关键问题

- 程序流程图



程序演示



The image shows a screenshot of a Microsoft Visual Studio debug console window. The title bar at the top reads "Microsoft Visual Studio 调试控制台". The console contains the following text, which represents an FTP session:

```
FTP>Control Connect...
220-FileZilla Server 0.9.60 beta
220-written by Tim Kosse (tim.kosse@filezilla-project.org)
220 Please visit https://filezilla-project.org/
FTP>USER:FANG
331 Password required for fang
FTP>PASS:***
230 Logged on
FTP>LIST
150 Opening data channel for directory listing of "/"
drwxr-xr-x 1 ftp ftp          0 May 22 11:48 A
drwxr-xr-x 1 ftp ftp          0 May 22 11:48 B
226 Successfully transferred "/"
FTP>QUIT
221 Goodbye
```


程序演示

服务器端日志

```
(not logged in) (127.0.0.1)> Connected on port 21, sending welcome message...
(not logged in) (127.0.0.1)> 220-FileZilla Server 0.9.60 beta
(not logged in) (127.0.0.1)> 220-written by Tim Kosse (tim.kosse@filezilla-project.org)
(not logged in) (127.0.0.1)> 220 Please visit https://filezilla-project.org/
(not logged in) (127.0.0.1)> USER FANG
(not logged in) (127.0.0.1)> 331 Password required for fang
(not logged in) (127.0.0.1)> PASS ***
fang (127.0.0.1)> 230 Logged on
fang (127.0.0.1)> PASV
fang (127.0.0.1)> 227 Entering Passive Mode (127,0,0,1,208,169)
fang (127.0.0.1)> LIST
fang (127.0.0.1)> 150 Opening data channel for directory listing of "/"
fang (127.0.0.1)> 226 Successfully transferred "/"
fang (127.0.0.1)> QUIT
fang (127.0.0.1)> 221 Goodbye
fang (127.0.0.1)> disconnected. |
```

本章小结

- **设计目的**
 - **FTP服务、FTP编程方法**
- **相关知识**
 - **应用层基本概念、FTP服务基本概念、FTP服务工作原理、FTP命令与应答**
- **例题分析**
 - **建立控制连接、登录FTP服务器、执行LIST命令**