***Project report***

# Design and implementation of FTP Client

**Course Title:**  *Internet Application*

**Name:** *Wang Zijun 2016213276*

*Lu Zelin 2016213353*

**Date:** 2019.6.6

**1. Overview**

This project is a simple user-friendly command line FTP client implemented by C language. The functions of this program contain upload and download files, get the file list, switch directory, transmission speed limit, breakpoint continuation etc.

## 2. Requirements Analysis

The requirement of the project is analyzed here, including,

* Development Environment, such as OS and programming language
* Functional requirements in details, including task decomposition and analysis

## 3. Preliminary Design

### 3.1 functional modules

* Utilities //useful peripheral functions
* Input handler //handle all input either from user or FTP server response.
* Transmit Function Implementation // concrete implementation of serval core function such as upload and download files, get the file list, limited speed download.

### 3.2 Interfaces

#### Utilities:

* int cliopen (char \* host, char \* port); //Establish a TCP connection to a server with a given server name and port
* void strtosrv(char \* str, char \* host, char \* port);// extract IP and Port from a string separated by “,”, e.g. given 192,168,1,1,23,43, then get host=”192.168.1.1”, host=”5931”
* int startListening(int controlFd);
* char \*getFileContent(char \*fname,int \*length);
* void getParaName(char \*str, char \* paraName,int \*length); //get first parameter of a command
* void getSecondParaName(char \*str, char \* paraName,int \*length);//get second parameter of a command

#### Input handler:

* void cmd\_tcp(int sockfd); // handle all input either from user or FTP server response.

#### Transmit Function Implementation:

* void ftp\_list(int sockfd); // send file list to command line
* void ftp\_get(int sck, char \* pDownloadFileName\_s); //download a file from remote host and write it in a local file
* void ftp\_put(int sck, char \* pUploadFileName\_s);//upload a local file to FTP server
* void ftp\_get\_limit(int sck, char \* pDownloadFileName\_s,int speed ); // download a file with a given limited speed from remote host and write it in a local file
* void ftp\_put\_limit(int sck, char \* pDownloadFileName\_s,int speed ); //upload a local file to FTP server with given speed.
* void ftp\_reget(int sck, char \* pUploadFileName\_s); // used to retransmit, download a file from a given offset point

### 3.3 Flow Chart



## 4. Detailed Design

#### Function Interpretation

##### int cliopen(char \* host, char \* port)

Given the host and port number in ascii form, this function will open a TCP connect with the port. This function will be used to open a control connection or data connection.

##### void ftp\_get(int sck, char \* pDownloadFileName\_s)

This function will firstly open a file descriptor, the continuously receive content from a given socket of the data connection of FTP server until the read no byte from the socket. Finally, the file stream and data socket will be closed by *fclose()* and *close()* respectively.

##### void ftp\_reget(int sck, char \*pDownloadFileName\_s)

This function get file from a break point. Firstly a existed file will be open with the “append” method. then it will receive byte stream from a offset point of the data connection socket *sck.*

when all residual data have been successful receive and write to local file, the file stream and socket will be close.

##### void ftp\_put(int sck, char \* pUploadFileName\_s)

This function firstly read the file from local host, then write the content in the given data connection by write(sck, file\_content, length), finally close the data connection by *close(sck).*

##### void ftp\_get\_limit(int sck, char \* pDownloadFileName\_s,int speed )

This function is used to retrieve a file from server with limited speed. Besides, the Interrupt download function also include in this function. Giving a limited speed( bytes/s), the according transmit buffer size will be calculated, then read content from data connection socket incessantly while limiting writing speed to local file by calculated buffer size and a fixed sleep interval.

##### void strtosrv(char \* bematch, char \* host, char \* port)

Extract IP and Port from Ftp server response, e.g. the server response is "227 Entering Passive Mode (10,3,255,85,42,90).\r\n" then use regular expression to get host=”10.3.255.85”, host=”10842”.

##### void getParaName(char \*str, char \* paraName,int \*length);

Get the parameter follow by the first command.

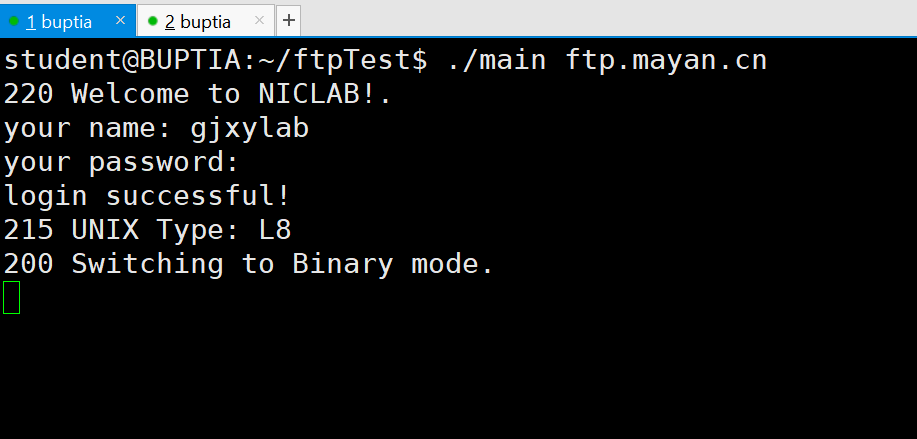
##### void getSecondParaName(char \*str, char \* paraName,int \*length)

Get the second parameter follow by the second command.

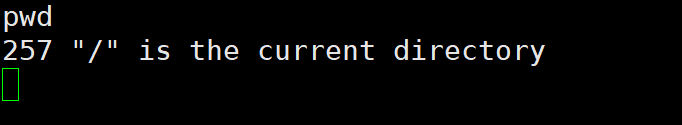
#### C:\Users\zijun\AppData\Local\Microsoft\Windows\INetCache\Content.Word\FlowchartDiagram1.jpgInput handler Module Flow Chart

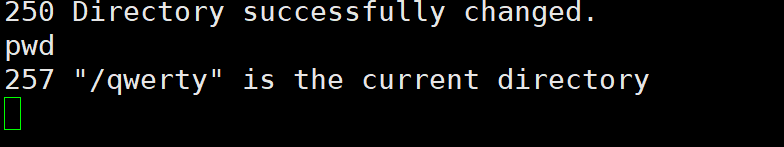
**5. Results**

### Login

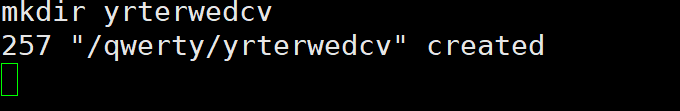


##### PWD

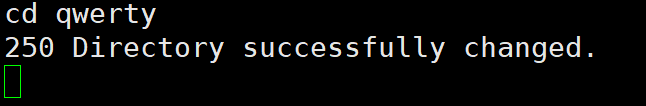




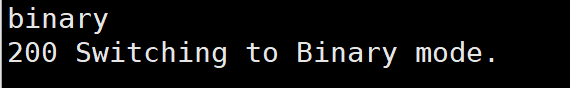
##### mkdir



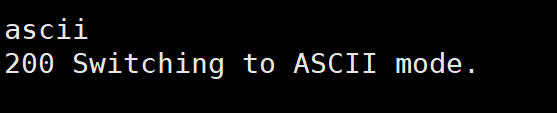
##### cd



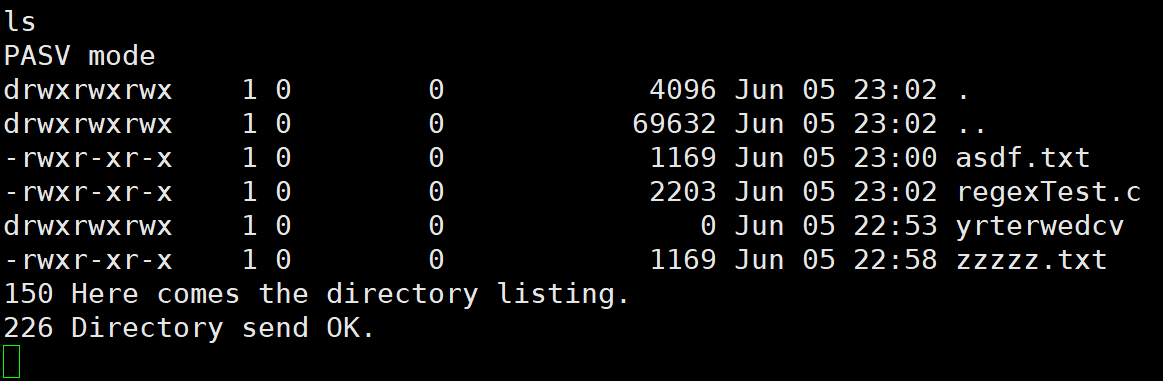
##### binary



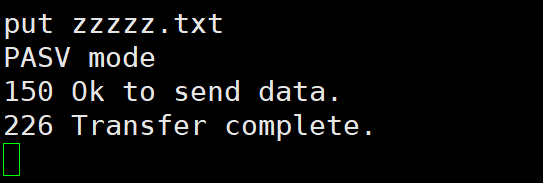
##### ascii



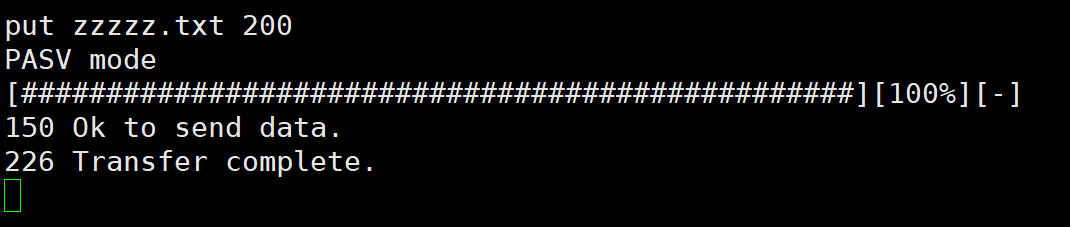
##### ls



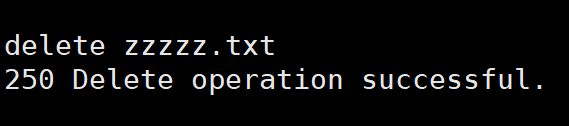
##### put



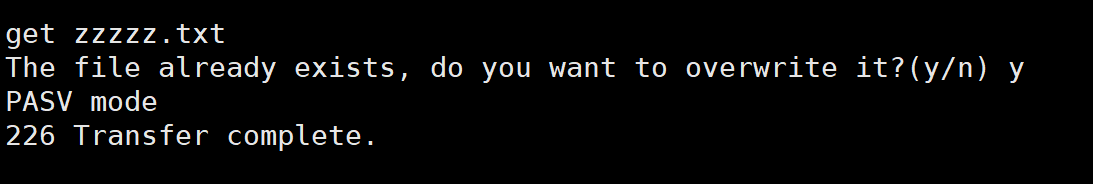
##### put with limited speed



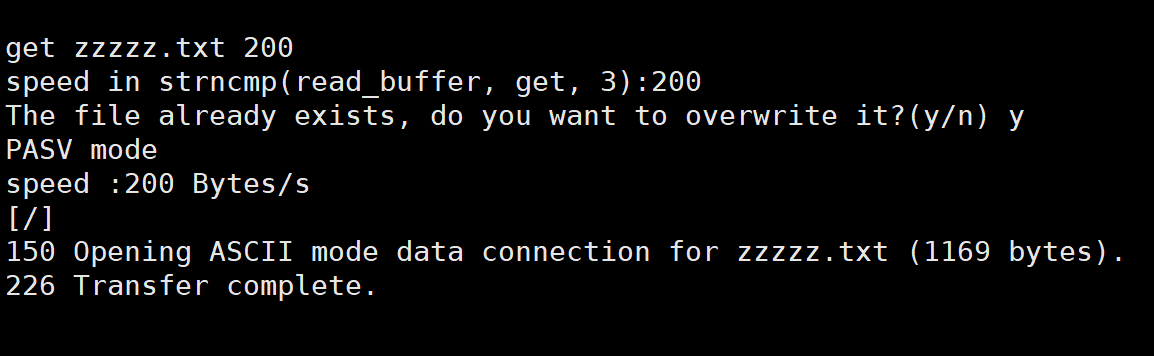
##### delete



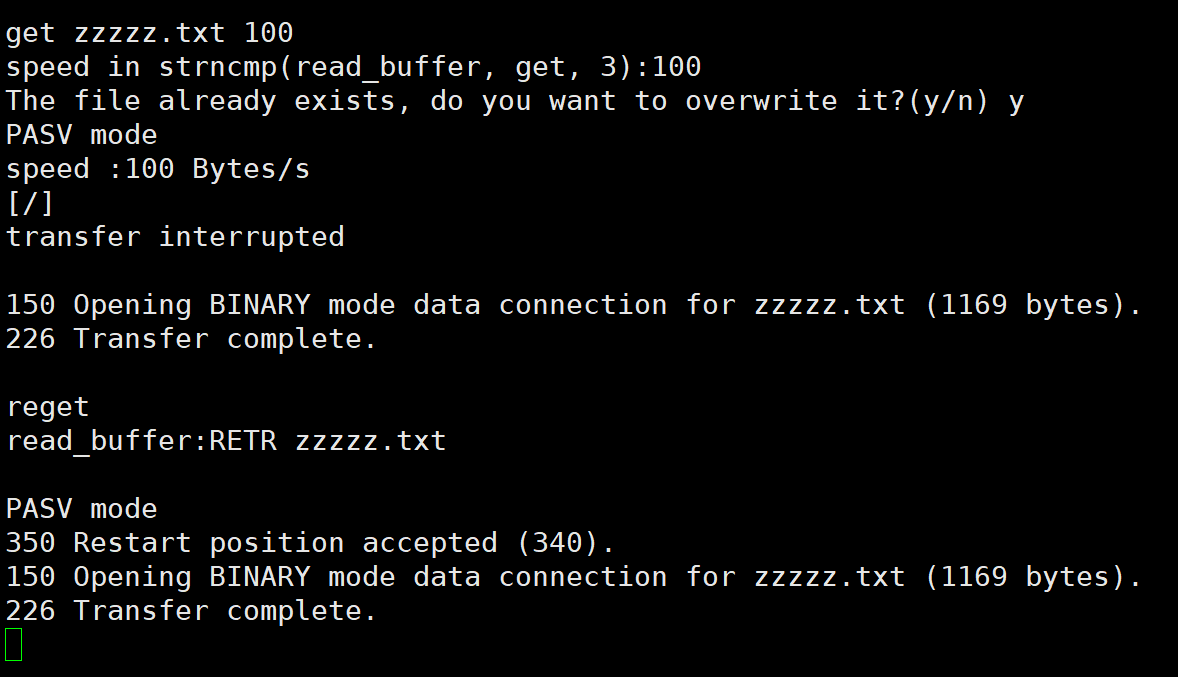
##### get



##### get with limited speed



##### interrupt get file, then reget from breakpoint



##### rename C:\Users\zijun\AppData\Local\Temp\1559749294(1).png

## 6. Summary and Conclusion

Specify how you divide jobs with your partner and your jobs in the project. Self-evaluation of your work is given here, including future improvement.

#### Work Division

Wang Zijun: User command and server response processing, put and get method implementation, additional functions.

Lu Zelin : socket programming, Implementation of the remaining instructions，String parsing

#### Self-Evaluation

We have a reasonable division of labor, the development process is very efficient, and all the basic functions and additional functions have been implemented.

#### Future Improvement

Increase the stability of the program, eliminate the possibility that the program is blocked, and add additional FTP commands.