程序运行说明文档

本项目实现了一个基于TCP协议的客户端-服务器应用程序。客户端读取指定的ASCII文件，将其内容分块发送给服务器。服务器接收这些块，并将每个块的内容反转后返回给客户端。客户端最终将所有反转后的内容写入输出文件。client和server都是基于TCP的命令行程序。client运行在host os，server运行在guest os。

一、运行环境：

Python版本：Python 3.7 或更高版本

依赖库：标准Python库 (socket, struct, time, random, string, threading)

1. 运行方法：

服务器：python server.py /python3 server.py

客户端: python client.py

1. 配置选项：

服务器：在server.py文件中配置服务器的IP地址以及端口

客户端：在运行时提示输入IP地址、端口、需要反转的文件地址、数据块的最小长度、数据库最大长度

1. 工作流程：

服务器：

1.服务器启动并绑定到指定的IP地址和端口。

2.服务器进入一个无限循环，等待来自客户端的报文。

3.对于每个接收到的报文，创建一个新线程来处理响应。

4.首先分析报文的前两个字节，得到接收到的报文类型

5.如果报文类型为1，解析后四个字节为需要接收报文的数量；然后向客户端发送2Bytes的仅包含报文类型为2的agree报文。

5.如果报文类型为3，解析后四个字节为接收数据的长度length，然后将数据进行解码并反转。将报文类型4，数据长度length，反转后的内容编码打包后形成reverseResponse报文，发送reverseResponse报文。

6.最后关闭当前套接字的链接。

客户端：

1.客户端启动并提示用户输入服务器IP地址和端口、需要反转的文件地址、数据块的最小长度、数据库最大长度

2.读取指定的ascii文件并依据最大、最小块长进行分块

3.向服务器端发送Initialization报文，给出发送的数据包数量N。收到agree报文后开始发送数据。

4.客户端发送N个数据包到服务器，每个数据包包含类型Type=3，数据内容长度、需要反转的数据内容。并接收反转后的数据包，解析后输出。

5.发送并接收完这些数据块后，将收到的反转后所有内容写入到新文件。

6.关闭套接字连接