

**计算机网络课程设计**

**报告**

实习任务名称： Project 2 实习报告

学 号： 20171001561

姓 名： 周麟

**中国地质大学地理信息工程学院**

**软件工程系**

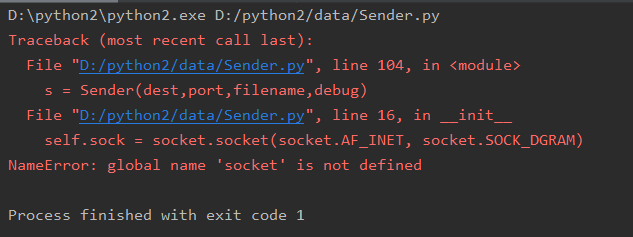
**2020年 5 月**

**1.运行情况：**

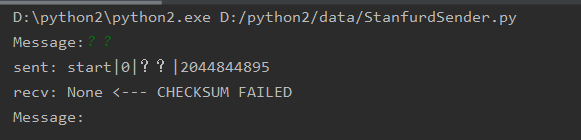
刚刚运行四个sender时， BasicSender，StanfurSende，UnreliableSender能运行，因为BasicSender没有主函数运行，只有类，仅仅起着框架的作用。StanfurSende，UnreliableSender虽然能运行，能发送数据，但是接收不到应答的消息。

无运行Receiver的情况：

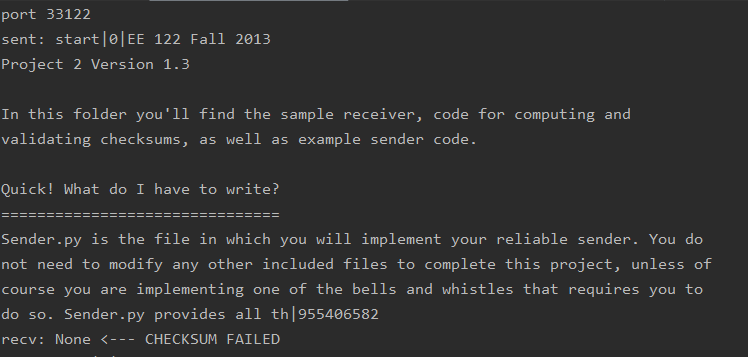
Sender：



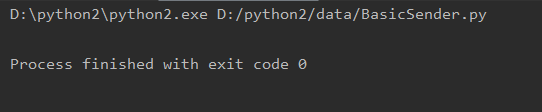
StanfurSender：



UnreliableSender：



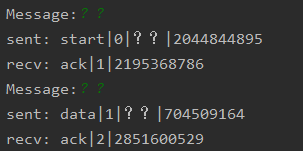
BasicSender：



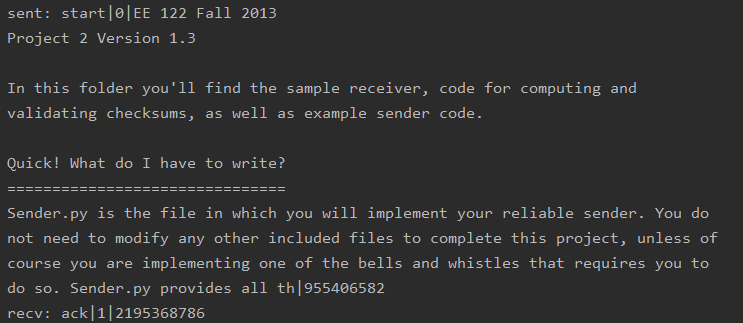
可以发现，StanfurSender和UnreliableSender的recv都是none，说明返回过来的信息表示没有收到包。

当运行Receiver之后，StanfurSender和UnreliableSender就可以发送包了。原因是如果Receiver没有运行的话，这两个Sender就无法连接发送数据。

StanfurSender：



UnreliableSender：



可以看到发送成功返回了ack。

而且生成了两个文件：



里面保存这发送的数据。

但是sender依然不能运行。因为sender的start函数没有实现。

**2.发送大文件：**

因为发送数据有限定的大小规定，所以如果想发送大文件就必须分包传送。现在要使sender能够运行，就在sender类里实现操作。

目前需要实现的操作有：

1. 输入发送的指定的文件名
2. 读入指定文件数据
3. 打包指定文件数据

1.1-2步骤都是open ‘rb’文件操作。

2.3步骤主要是每次只读一定量的文件数据发送。当文件数据为0时，标记为end，表示发送完毕。

在class sender中的start添加代码和加入了一个Response消息的解析。

代码：

def start(self):

seqno = 0

msg = self.data.read(1000)

msg\_type = None

while not msg\_type == 'end':

next\_msg = self.data.read(1000)

msg\_type = 'data'

if seqno == 0:

msg\_type = 'start'

elif next\_msg == "":

msg\_type = 'end'

packet = self.make\_packet(msg\_type,seqno,msg)

self.send(packet)

print "sent: %s" % packet

response = self.receive()

self.handle\_response(response)

msg = next\_msg

seqno += 1

self.infile.close()

def handle\_response(self,response\_packet):

if Checksum.validate\_checksum(response\_packet):

print "recv: %s" % response\_packet

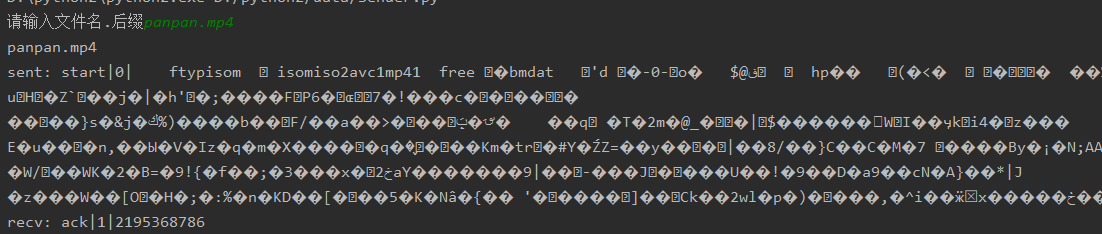
else:

print "recv: %s <--- CHECKSUM FAILED" % response\_packet

如果想发送一个视频：

比如：





之后就会生成一个文件：



和发送的视频文件大小相同。

打开方式用视频软件打开。