IERG 3310 Project lab report

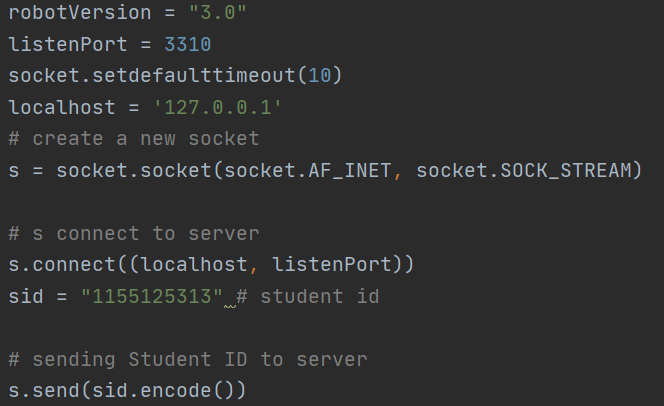
Name: Cheng Wing Lam

SID : 1155125313

Steps 1: Run Robot.exe (It is already given)

Step 2:

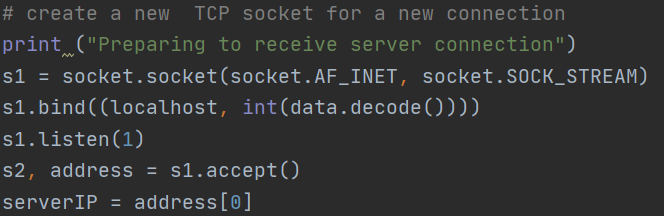
I make a new TCP socket first and make connection. After Student connects to Robot, call send() to send 10 char student ID to Robot.



Step 3:

Robot will send 5 char string to Student. So, I need to call recv() to receive the string in Student.



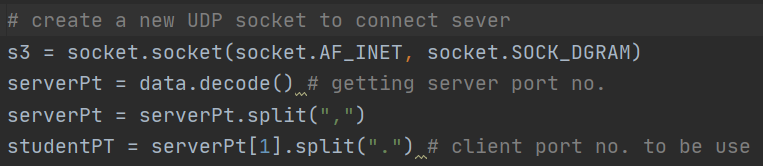
After received the 5 char string. I use this string to create a new TCP socket s1 for new connection.

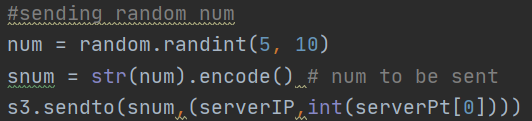
Step 4:

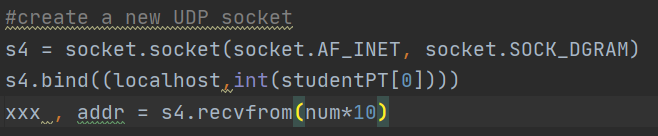
The Robot sends 12 char string “fffff,eeeee.” So, I call recv() in Student for receiving.

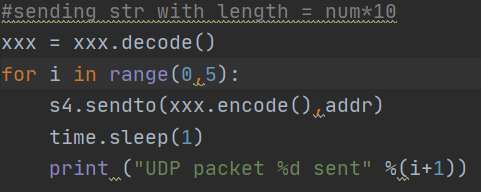


Then I decode the ‘data’ to get the string and use it to create a new UDP socket s3.



Then I generate a random number num ( 5 < num < 10) and send it to Robot.Then, I create a new UDP socket s4 with port no. eeeee to receive the num \* 10 length string from Robot.

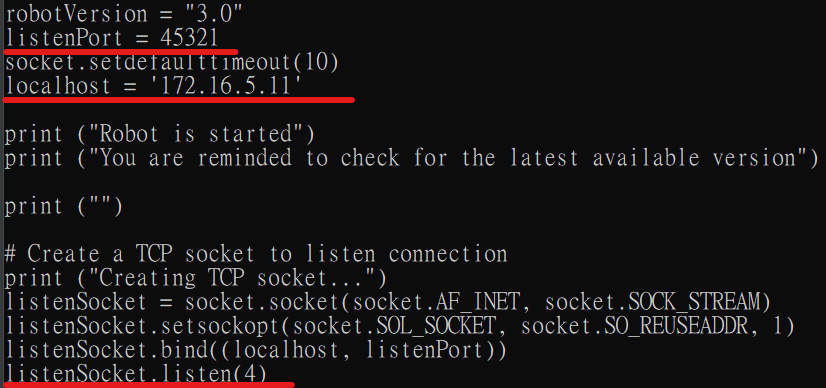


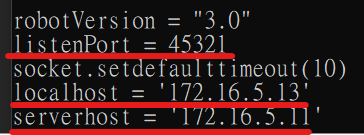
Step 5:

I call sendto() to send the string 5 times to ROBOT at UDP port fffff.

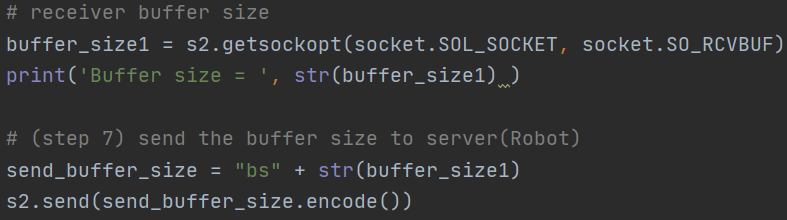
Step 6:

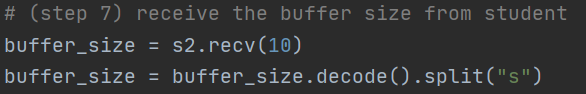
I need to run Robot and Student in two different machines. So, I make some arrangement about the IP ,port no. and listen(). Other code is same as step 1-5.

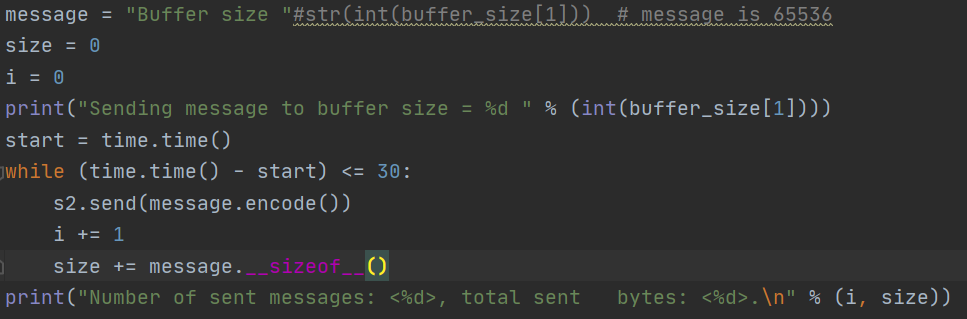
Robot : Student :



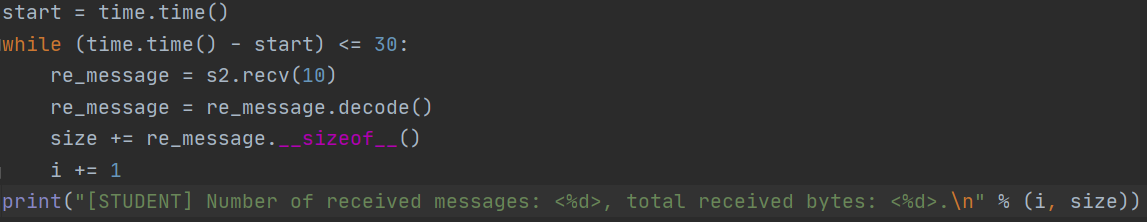
Step 7:

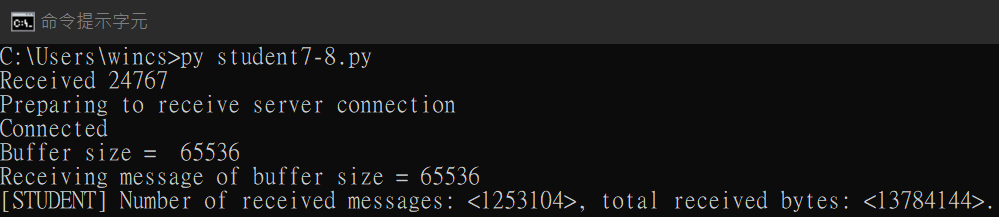
First, get the receiver buffer size of Student and send it to Robot

To receive buffer size, call recv() again

Then make a while loop to send a large number of messages to Student within 30 seconds

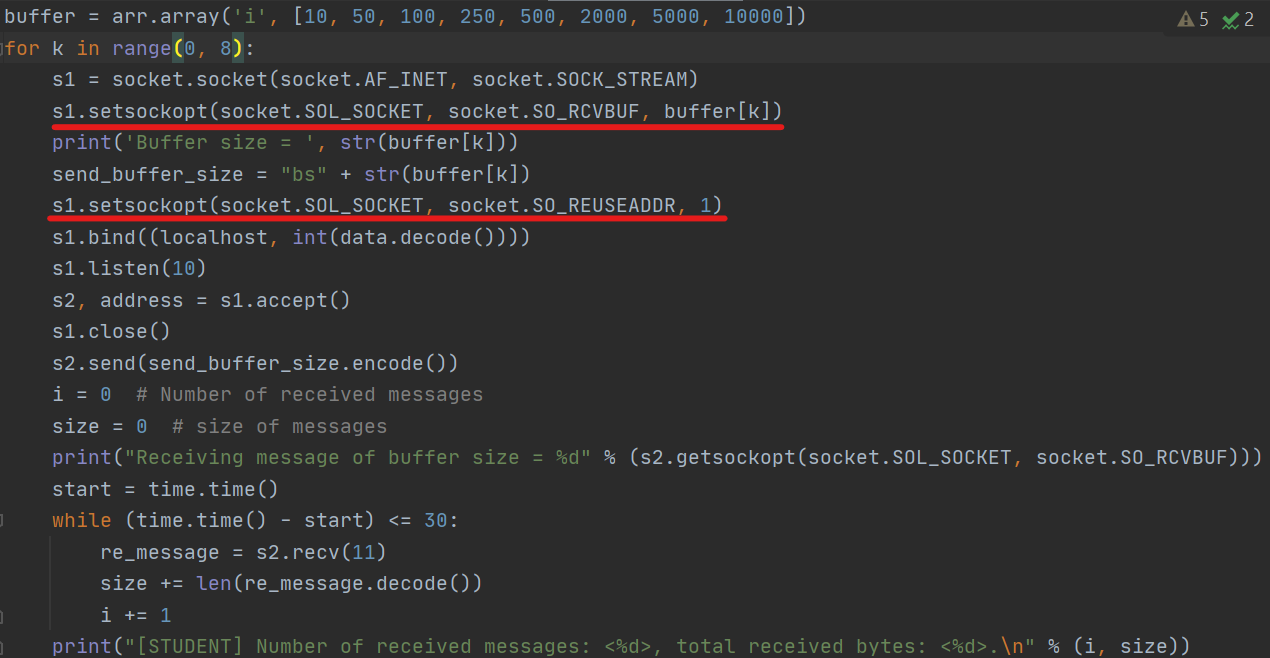
Student also set a while loop receive message and count how many message is received in Student



This is the result

Step 8:

Same as step 7, just adjust the receiver buffer size into [10, 50, 100, 500, 2000, 5000, 10000]% bytes. So, I add a while loop for this. (This is student. Robot is similar) Also, need to reset the socket that the address and port no. can be reused.



Change the buffer size here

Make the socket that can be reused

|  |  |  |
| --- | --- | --- |
| buffer size | total bytes | system throughput |
| 10 | 10689810 | 356327 |
| 50 | 31293801 | 1043127 |
| 100 | 14022459 | 467415.3 |
| 250 | 13600246 | 453341.5 |
| 500 | 13673649 | 455788.3 |
| 2000 | 13768777 | 458959.2 |
| 5000 | 13384481 | 446149.4 |
| 10000 | 13515216 | 450507.2 |

This is the throughput

We can see that as the buffer size increase, the system throughput also increases. Also, if the message size is changed, the optimal receiver buffer size is changed to. In this case, my message is “buffer size”, the message size is 11. So, 50 is the optimal buffer size in this situation.

What is the limitation of the experiment set-up since we do it in IE Common Lab? (Hint: What if the network is a low-speed one? )

Some packet may be lost. It makes Student received less packets and affect the throughput. so that, the throughput results are not only affected by the buffer size. The network connectivity also affects the throughput. Therefore, we cannot find the optimal receiver buffer size.