Socket Programming Project



**CS 334/CS 534 Lab #4**

**:**

**Socket Programming Project**

**Due on 11:59pm, October 14, 2017**

# Project Requirements

In this project, you will gain hands-on experience on both TCP and UDP socket programming. you are required to write a program (STUDENT) to interact with the provided program (ROBOT). You are allowed to use the ROBOT source code given (either Java or Python) in the project package to complete this project. All students should finish Steps 1 to 6. This project can be completed by a group of two students or alone. It is your responsibility to find your teammate if you choose to work as a group. You will need to demo your program to the TA as required by Step 6, and if you work alone, the TA will run the ROBOT program on his machine to help you demo your STUDENT program.EVERY student should turn in their own source code whether they work in a group or not.

This project will be conducted over two lab sessions. Lab #1 is on September 23, 2016, and will teach preliminaries about the lab tasks and programming basics in Python 2.7. Lab #2 is a working lab and where you will demonstrate the finished program to the

TA, and it is on October 14, 2016. Each group should compile, run, and demonstrate their program in front of the TA. The TA may ask some questions related to the program, answer correctness will affect your grading of this lab. Before the end of your lab session on October 14th, you must demonstrate your program to your TA in order to receive full credit.

**Steps:**

Your program has to interact with ROBOT according to the following steps:

1. Start the ROBOT by running “robot.exe” in a Windows command prompt.

1. When the ROBOT is started, a message “ROBOT IS STARTED” will be printed, indicating that the ROBOT is now listening on TCP Port 3310. STUDENT has

to connect to the ROBOT TCP Port 3310 and send your BlazerID via the connection established.

1. The ROBOT will then send a 5 char string *ddddd* to the STUDENT. STUDENT will need to create a TCP socket s\_2 at port *ddddd* to accept a new connection.

The ROBOT will initiate the new connection 1 second later after sending *ddddd*.

STUDENT

ROBOT

Connect

Send student ID

Recv

***5***

char string dddddd

.

SOCKET s1

SOCKET s\_2 listening

for connection on TCP

ddddd and get s2

Wait 1 second before

making new connection

to port ddddd

Listening on

TCP 3310

Accept the new TCP connection

Upon accepting the connection, a new socket s2 will be returned.

1. ROBOT will then send a 12 char string “*fffff,eeeee.”* to the STUDENT using the new connection. STUDENT needs to decode the message and create a UDP socket s3 to send a variable ***num*** ( 5 < ***num*** < 10) to ROBOT on port *fffff*. Then ROBOT will send a char string *xxx* with length ***num \* 10*** to STUDENT one second after receiving ***num***  and STUDENT will receive the string using s3 on port *eeeee* .

## SOCKET s2

STUDENT

ROBOT

Recv 12 char string fffff,eeeee.

Sendto

variable

***num***

Recv

***num\* 10***

char strin

g xxx

.

UDP port: fffff

SOCKET s3 on

port: eeeee

After

receiving

***num***

,

wait for 1 second to

send char string xxx

ROBOT will send the string *xxx* 5 times, once every 1 second. STUDENT only needs to receive any one of them.

1. When the STUDENT received the char string *xxx*, it will send back the string to the ROBOT at UDP port *fffff*. Similar to the ROBOT, the string will be sent 5 times, once every 1 second. The ROBOT will check if the two strings are the same.

STUDENT

ROBOT

Sendto

***num\*10***

char string 5 times

SOCKET s3

UDP port: fffff

***num\*10***

char string

1. In this step, you need to run ROBOT and STUDENT on 2 different machines and make sure all the steps above can be successfully executed. You may need to use Wireshark on the machines for network level debugging if necessary, e.g. if there is a firewall on either of the machine and/or along their end-to-end path, the programs may fail. It is then your responsibility to fix it, e.g. by choosing a different pair of machines elsewhere and/or configure them properly. Demo your completed project to the TA.

(For steps 1 to 6, the original provided ROBOT will be used to test your STUDENT program.)

# Programming Language and Platform

Students can use Java or Python to complete the project. The Networking Lab is equipped with Python 2.7. If a student wishes to use Java, he/she must use his/her own computer. If a student uses his/her own computer, he/she is required to be able to connect his/her computer to the network in the lab via an Ethernet cable.

# Submission

Students are required to submit a zip file containing the **executable files** and **source code** of both STUDENT and ROBOT.

**Project Submission Checklist:**

* The final source code for STUDENT and ROBOT and an executable file for each of them.
* If you worked as a group, please comment the first line of your code about your group member information:

o (For Python) #My Name: XXX, My Partner: XXX o (For Java) //My Name: XXX, My Partner: XXX

* If you worked alone, please comment the first line of your code with “I XXX Worked Alone”

Please kindly clear the history and the work (any screenshots/documents) you have done so far from the lab machine before you log out and leave the lab classroom the lab. Thank you.