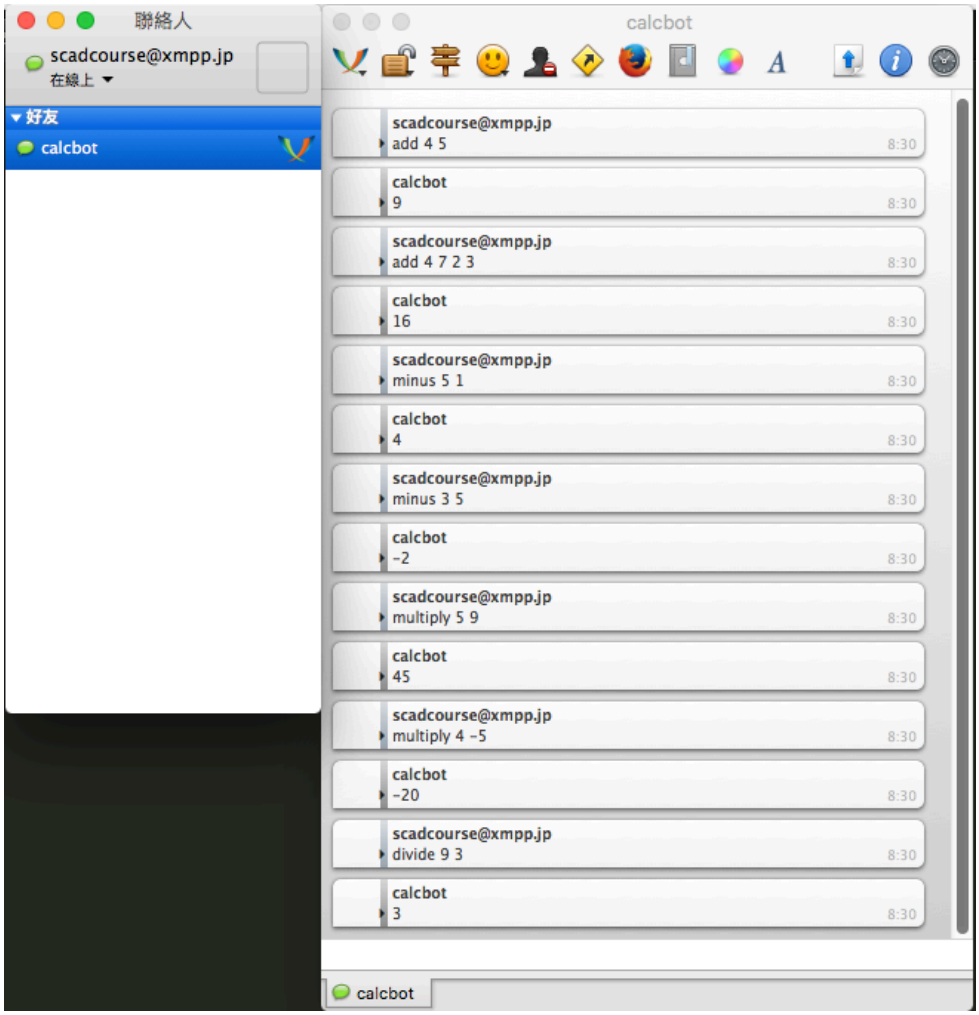
Social Computing Application Design
Assignment 2: Instant Messaging

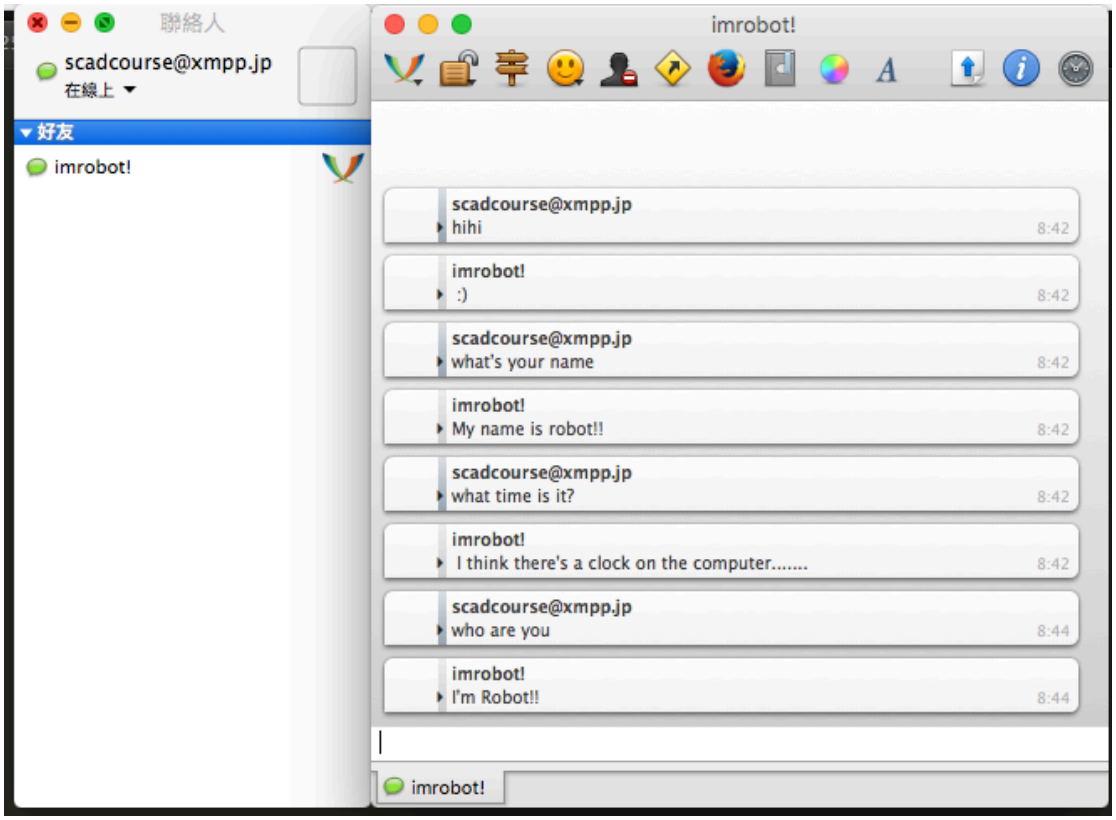
Honor Code

Any cheating will be handled seriously in compliance with the university rules. All assigned work is expected to be individual, except where explicitly written otherwise (e.g., term project). You are encouraged to discuss with your classmates; however, what you hand in should be your own work.

1. Write a simple IM bot in python that interacts with users. By asking your IM bot elementary math questions, a user should get the correct answers from it. You only need to implement 'add', 'subtract', 'multiply' and 'divide' commands for this calculator bot. Screenshots below show how users interact with this calculator bot. Bonus credits may be given to extra work.



2. Write a chat bot that can chat with users in a naturalistic manner. You are not required to perform sophisticated language processing. Simply scan for keywords in the input and pull out the response that matches the keywords the most. You are encouraged to use your knowledge and creativity on designing the mechanism. The following screenshot shows an example.



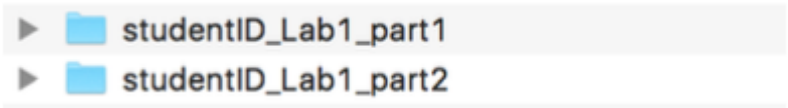
To ensure the scalability of your code, you are expected to store your keywords and corresponding responses in a text file. Whenever you start your chat bot, the program should load the file to obtain the set of candidate responses. You can take a look at the sample txt files we provided. Then you can either imitate the style or create your own ways to represent the data. (Note: sentences.txt contains regular expressions to identify the sentence patterns of the user input. You are encouraged to try using regular expressions to make your robot smarter.)

Hints:

Don't reinvent the wheel. Use third-party libraries can help you focus on designing your application. You can reuse the **sleekxmpp** library code that we've talked about in the lab to complete this assignment. You can check the lab slides again for details of the library.
You can also check https://docs.python.org/3/library/re.html for python's regular expression operations.

Notices:

1. **Deadline: 2015/10/25 23:59 (If you submitted in 10/26 00:00-23:59, you will get partial credits (70% of the original score). No credits if submitted afterward.)**
2. To submit your assignment, you should follow the form like below:

   Put the source code, Readme.txt in each part of the assignment. Then zip each part to a **zip** file named as "id_Lab2.zip" and upload to iLMS.
3. For each source code file, you have to add comments to explain your code.
4. The Readme files have to include the brief explanation of your work (how you implement such features, the running environment of your code), the problems you encounter and how to solve the problems. Below is an example: