Project2: rwg Systems

NP TA 孟宇

11/18 23:59

Project 2 Deadline.

Demo: 11/19 Tue.

rwg - Remote Working Ground

- Chat-like system
- Provide all functions in **project 1**
- New functions
  - **User pipe**
  - **who -** get information of all users
  - **name -** rename
  - **tell -** send message to someone
  - **yell -** broadcast message

3 Servers

- np_simple (Single user)
  - **Project1**
  - Concurrent connection-oriented
- np_single_proc (Multiple users)
  - **Project1** + **User pipe** + **4 Functions** + Broadcast message
  - Single-process concurrent
- np_multi_proc (Multiple users)
  - **Project1** + **User pipe** + **4 Functions** + Broadcast message
  - Concurrent connection-oriented + Shared memory

Project 2: Submission

- Create a directory named as your student ID, put all files into the directory.
- You must provide Makefile. Three executable files named **np_simple** (server 1), **np_single_proc** (server 2) and **np_multi_proc** (server 3) should be produced after typing make command.
- You are NOT allowed to demo if we are unable to compile your project with a single make command.
- Upload only your code and Makefile. Do NOT upload anything else (e.g. noop, removetag, test.html, **.git**, **__MACOSX**)
- zip the directory and upload the .zip file to the E3 platform.
ATTENTION! We only accept .zip format

About Testing

- We will provide our client program and demo script. Before that, you can use **telnet** to test your server
- **Only one** user enter command at one time
- No inevitable race condition
- No space at the end of commands

Testcase Example

login1
login2
login3
1:name Pikachu
3:name Rayquaza
3:who
2:ls | cat | cat | cat | cat | cat | cat | cat | cat | cat | cat | cat | number
3:yell Find hotel
2:yell Trivago
...

Invalid Testcases

- % ls >2 | number (two output)
- % ls |1 number <1 (two input)
- User pipe to himself/herself
- % cat <1 >2 , but >2 cause error
- Wrong arguments for build-in functions
  - % tell hahaha

Important Details

- **All ' are ' on spec , do not copy directly**
- Set flag **SO_REUSEADDR** and **SO_REUSEPORT** in server socket
  - Hint: use function **setsockopt**
- All servers should listen on the port assigned by the first argument
  - **./np_single_proc 12345**
- Listen for connection again on np_simple (server 1) after user exit
- Handle Signal **SIGINT** to clean shared memory

About Demo

- Correct format and compile

- QA

- Pass np-basic test cases

- Pass np-hard test cases

- Implement 1 or 2 extra functions with limited time

Remind

- You can only implement this project with **C and C++**, other third-party libraries are **NOT allowed**.
- Start project2 earlier
- You are HIGHLY encouraged to publish your questions on E3 討論區
- Don't leave any zombie processes in the system
- **We take plagiarism seriously**
- Protect your code from being stolen
- Enjoy the project !!