Auto-Grading

Use Java socket to implement a much simplified software testing tool to grade submitted homework. In the following, *user* refers to a person who interacts with the client program of this grading application.

The client program uses a JTextField to receive from user the IP address of a server. Once the connection with the server program is established, it receives from server the Page.class file, loads the file, and displays in its own frame, the content of this page. The content of this page includes GUI components to allow a user to upload a file together with his/her name.

The users submit their C source code to the server, along with their names. For each submission, the server should perform automated testing and send the testing report back to the user. In addition, the server should create a separate directory, under the current directory, with the name of the user. In this directory, the server should record the submitted source code and the testing report in two separate files.

The name of a test case file should be a command line argument to the server program: the same server program can be started multiple times with different test case files for checking different homework, e.g. test case file #1 for homework #1, test case file #2 for homework #2. Once started, the server should keep receiving submissions for the same test case file.

For each submission, the server program should first check whether the code can be compiled successfully. The result should be part of the testing report.

If the submitted work can be compiled successfuly, the server should run the executable file generated from the submitted code against the test cases given in the test case file. For each test case, the testing report should include the result whether this test case is passed or not.

The C code submitted by the users has this property: it uses standard I/O to get several lines of input, and then gives one line of output.

The test case file has test cases separated by a line starting with a #. Each test case contains several input lines followed by an output line separated by a line starting with a *.

For each test case, the server program creates a new process to run the executable file of the submitted code, and to check if the output conforms to the expected one. For each test case, the executable from user's submission is restarted. The server program, on the other hand, goes through all test cases for each submission, for all the submissions, without termination.

Sample files are provided to test your program. *testcase1.txt* is a test case file for *nocompiler.c*, *sample1a.c*, *sample1b.c*. *testcase2.txt* is a test case file for *nocompiler.c*, *sample2a.c*, *sample2b.c*.

You can use the following assumptions for simplicity:

- Each user can submit one and only one file, with file extension *.c*.
- The submitted code does not perform file I/O.
- You can use Control-C to terminate server program, so the server program can simply run in an infinite loop to keep receiving submissions.
- The users will submit their work one by one in the sense that the second submission will not start until the first one has received testing report. With this assumption, you do not have to handle simultaneous submissions to the server.

Your code should work properly when client and server programs are executed on different machines.

To submit this grading assignment, upload three source code files called Server.java, Client.java, and Page.java. Your Page.java should have a String variable called *serverip* together an assignment statement like *serverip = "xxx.xxx.xxx.xxx";* which should be the only point of necessary change in order to run your program with a different IP address of the server program.