

Assignment 1: Distributed File Storage

Due date: Thursday, October 20, 2016 at 22:00

This is an individual assignment. You may discuss it with others, but your code and documentation must be written on your own. You might receive a bonus for an outstanding solution.

Implement a distributed storage system consisting of a number of servers each managing a set of files on a host. A user can use a controller application to list the files stored within the system, and to instruct a server to copy a file from a server to one or more other servers. The server should be able to serve more than one request at a time. Specifically, you must implement two independent Java applications. One, called *FileController*, implements the controller component. The other one, called *FileServer*, implements the server component. The *FileController* application must work as an interactive shell, reading commands from the standard input, with one command per line. At a minimum, *FileController* must support the following commands.

- A *list* command that takes a server address and lists the files managed by that server. A server address is a host name or address followed by a semicolon (':') and followed by a port number.

For example, the command

```
list 10.63.129.163:9999
```

must list the files managed by the server responding on port 9999 on host 10.63.129.163.

- A *copy* command that takes a source server address, a source file name, and one or more destination server addresses, and copies the given file from the source server onto all the destination servers.

For example, the command

```
copy 10.63.129.163:9999 image.jpg 10.63.129.155:1234 10.63.129.156:3421
```

must instruct the source server (10.63.129.163:9999) to copy its file “image.jpg” to destination servers 10.63.129.155:1234 and 10.63.129.156:3421.

The *FileServer* application takes two optional command-line parameters. The first parameter defines the port number the server should use. The second parameter defines the directory in which the server manages files. The default directory is the current directory.

Notice that, in developing your file storage system, you must also design a protocol for the communication between a controller and a server, and also between a server and another server.

Submission Instructions

Submit all your source files. Add comments to your code to explain sections of the code that might not be clear. You may use an integrated development environment (IDE) of your choice. However, *do not submit any IDE-specific file*, such as project description files, and *make absolutely sure that the files you submit can be compiled and tested with a simple invocation of the standard javac compiler and the standard java virtual machine.*

In addition to the source files, submit a text file called *README* containing a brief description of your implementation, including a description of the communication protocol you developed, and also a list of all the limitations and errors you are aware of were unable to fix.

Package all the files you need to submit in an archive file named

```
a01-<lastname>-<firstname>
```

and submit that file through the iCorsi system.