Documentation

Since the program requirement has clearly explained the implementation for most of functionality of this program, this documentation will mainly focus on the algorithm implementation of FileServer that doesn't mentioned in the requirement.

General Description

The program FileServer provide the ability to interact with the FileClient. It runs as a server which maintain the file cache. It is composed with two main functions download() and upload() which are used to be called from FileClient, and a FileCacheEntry class which denotes the file entry in the memory. FileCacheEntry also has download() and upload() method which has the ability to manipulate the file which already cached in the list.

wait(), notify(), notifyAll(), writeback(), invalidation()

The suspend and resume of download is implemented with wait(), notify, and notifyAll() in the FileCacheEntry's download() and upload() method.
And the file consistency is implemented by the use of writeback() and invalidation() RMI call in the FileCacheEntry's download() and upload() method.

- **Solution of (5d)**

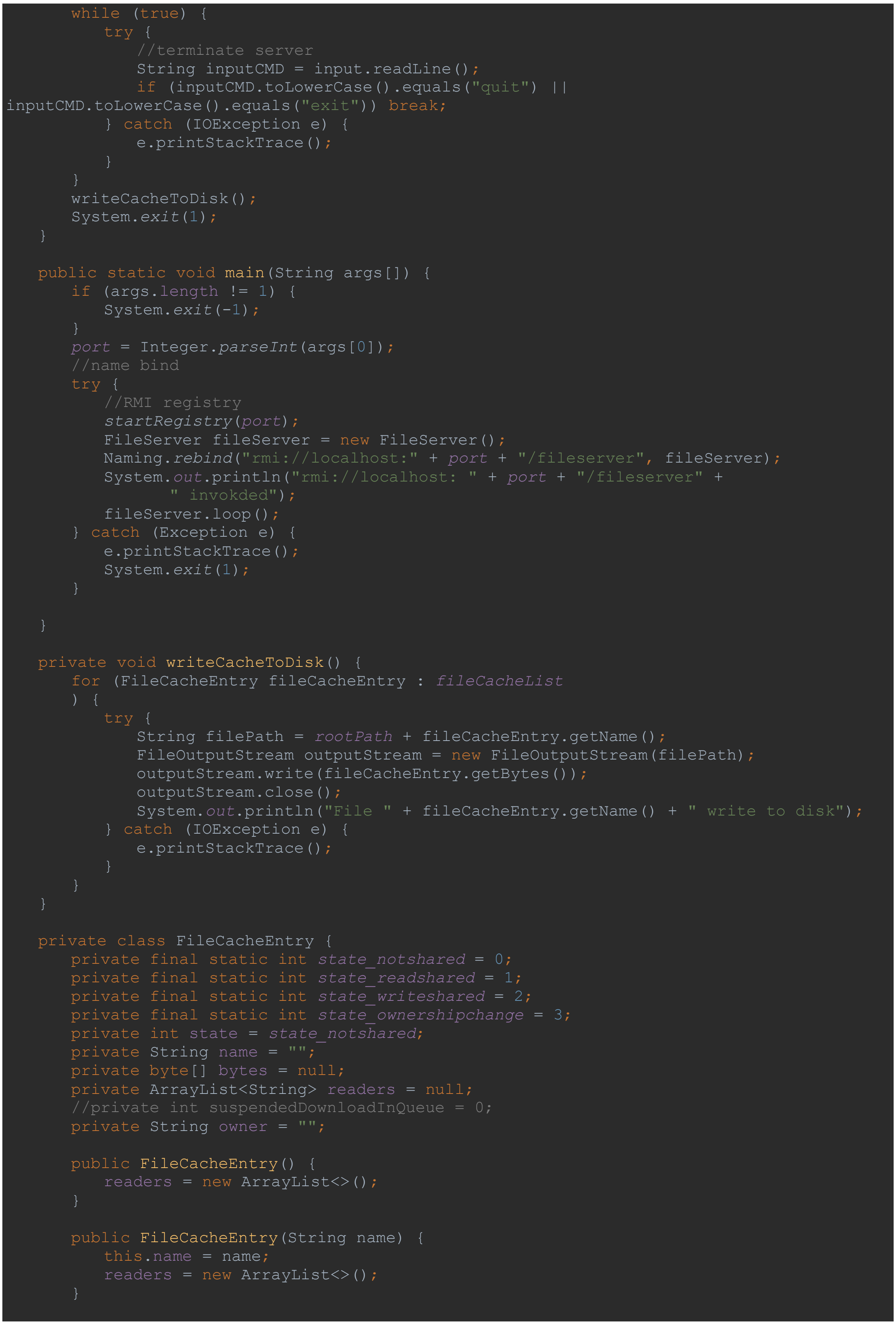
If a download write request is received on server when the file state is write_shared, it will set the state as ownership_changed first, and then send the RMI call of writeback to the current owner client, at last it will call wait() on itself to be suspended. After the current owner finish write, it would start running writeback() and it triggers upload() on FileServer. The FileServer will traverse the readers list and make the invalidation() RMI call to each of readers. Then it call notify() to wake up the suspend thread. When this thread got woke up, it will set the owner as itself also call notifyAll(), and the use of notifyAll() will be explained on the next section

- **Solution of (5f)**

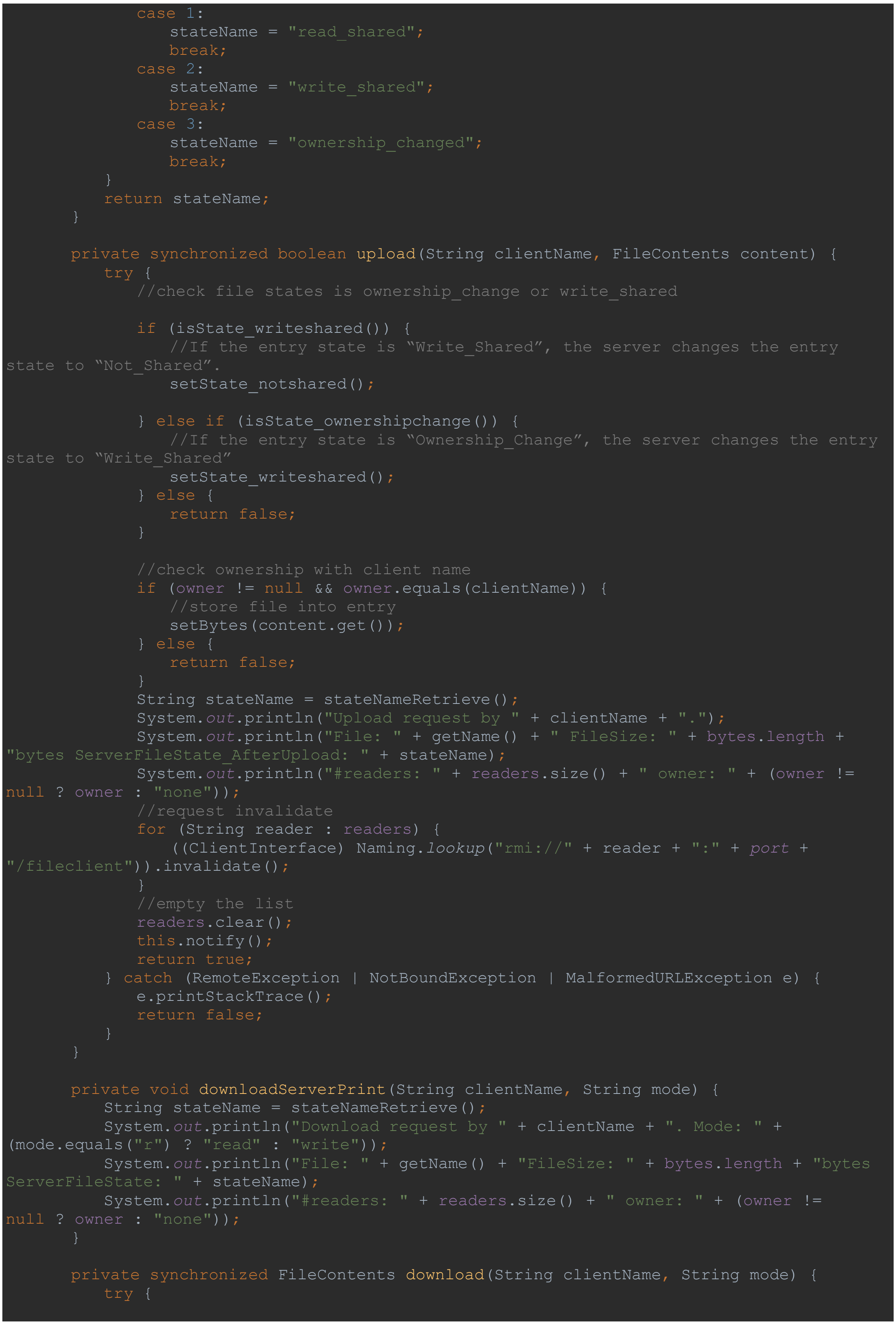
If a download request is received on server when the file state is ownership_changed, it will do nothing but call wait() at first, because the current thread cannot foresee which will be the next owner to send. After the thread which is downloading the file with write_shared that explained in the above section call notifyAll(), the current thread with file state ownership_changed will be resumed, and now it can make a writeback() RMI call to the current owner client(which is the client that request download on write_shared file state), and then it call wait() again to suspend itself. After the current owner client finished uploading and call notify(), this thread will wake up, then it set the state as write_shared, set the owner as itself, and call notifyAll(), this notifyAll() is called because there might be multiple of clients are currently suspend on ownership_changed status, so it can wake up one of the other thread as the next to make writeback() RMI call.

Source Code

```
import java.io.*;               // IOException
import java.net.*;              // InetAddress
import java.rmi.*;              // Naming
import java.rmi.server.*;         // UnicastRemoteObject
import java.rmi.registry.*;       // rmiregistry
import java.util.ArrayList;


public class FileServer extends UnicastRemoteObject
        implements ServerInterface {

    private static final String rootPath = "/tmp/";
    private static int port;
    private static ArrayList<FileCacheEntry> fileCacheList;

    protected FileServer() throws RemoteException {
        fileCacheList = new ArrayList<>();

    }


    public FileContents download(String client, String filename, String mode) throws
RemoteException {
        //scans the list of file cache entries
        boolean isInList = false;
        FileCacheEntry entry = null;
        for (FileCacheEntry fileCacheEntry : fileCacheList) {
            if (fileCacheEntry.getName().equals(filename)) {
                entry = fileCacheEntry;
                isInList = true;
                break;
            }
        }

        if (!isInList) {

            //if the server does not find a requested file’s cache entry in this list
            try {

                System.out.println("Download request by " + client + ". Mode: " +
(mode.equals("r") ? "read" : "write"));
                //reads the file contents into the entry from its working directory
                String filePath = rootPath + filename;
                FileInputStream file = null;
                file = new FileInputStream(filePath);
                byte[] bytes = new byte[file.available()];
                file.read(bytes);
                //create and add the entry to the list
                entry = new FileCacheEntry(filename);
                entry.setBytes(bytes);
                if (mode.equals("r")) {
                    //register client's IP name in the readers of this cache
                    entry.addReaders(client);
                    //changes the entry state from “Not_Shared” to “Read_Shared”
                    entry.setState_readshared();
                    System.out.println("File: " + filename + "FileSize: " + bytes.length +
"bytes ServerFileState: read_shared");
                    System.out.println("#readers: 1 owner: none");
                } else {
                    //register client's IP name in the owner of this cache
                    entry.setOwner(client);
                    //entry state changes from “Not_Shared” to “Write_Shared"
                    entry.setState_writeshared();
                    System.out.println("File: " + filename + "FileSize: " + bytes.length +
"bytes ServerFileState: write_shared");
                    System.out.println("#readers: 0 owner: " + client);
                }
                fileCacheList.add(entry);
                //store the cached file contents in a FileContent object and passes the DFS
client

                return new FileContents(bytes);

            } catch (IOException e) {
                //if file not found in disk
                if (mode.equals("w")) {
                    //create entry for the client if file not existed for write request
                    entry = new FileCacheEntry(filename);
                    // create file content as empty contents.length == 0
                    byte[] bytes = new byte[0];
                    entry.setBytes(bytes);
                    entry.setOwner(client);
                    entry.setState_writeshared();
                    fileCacheList.add(entry);
                    System.out.println("File: " + filename + "FileSize: " + bytes.length +
"bytes ServerFileState: write_shared");
                    System.out.println("#readers: 0 owner: " + client);

                    FileContents fileContents = new FileContents(bytes);
                    return fileContents;
                } else {
                    System.out.println("File " + filename + " not found in the list and
disk");
                    //returns "null" to the client if file not existed for read request
                    return null;
                }
            }

        } else {
            //if the server find the request file in the list
            return entry.download(client, mode);
        }

    }


    public boolean upload(String client, String filename, FileContents contents) {
        //find file entry in cache
        FileCacheEntry entry = null;
        for (FileCacheEntry fileCacheEntry : fileCacheList) {
            if (fileCacheEntry.getName().equals(filename)) {
                entry = fileCacheEntry;
            }
        }
        if (entry != null) {
            return entry.upload(client, contents);
        } else {
            return false;
        }
        //resuming the download( ) function that has tried to download the same file for a
write
    }

    private static void startRegistry(int port) throws RemoteException {
        try {
            Registry registry =
                    LocateRegistry.getRegistry(port);
            registry.list();
        } catch (RemoteException e) {
            Registry registry =
                    LocateRegistry.createRegistry(port);
        }
    }


    private void loop() {
        BufferedReader input = new BufferedReader(new InputStreamReader(System.in));
```



```
        while (true) {
            try {
                //terminate server
                String inputCMD = input.readLine();
                if (inputCMD.toLowerCase().equals("quit") ||
inputCMD.toLowerCase().equals("exit")) break;
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
        writeCacheToDisk();
        System.exit(1);
    }

    public static void main(String args[]) {
        if (args.length != 1) {
            System.exit(-1);
        }
        port = Integer.parseInt(args[0]);
        //name bind
        try {
            //RMI registry
            startRegistry(port);
            FileServer fileServer = new FileServer();
            Naming.rebind("rmi://localhost:" + port + "/fileserver", fileServer);
            System.out.println("rmi://localhost: " + port + "/fileserver" +
                    " invokded");
            fileServer.loop();
        } catch (Exception e) {
            e.printStackTrace();
            System.exit(1);
        }

    }

    private void writeCacheToDisk() {
        for (FileCacheEntry fileCacheEntry : fileCacheList
        ) {
            try {
                String filePath = rootPath + fileCacheEntry.getName();
                FileOutputStream outputStream = new FileOutputStream(filePath);
                outputStream.write(fileCacheEntry.getBytes());
                outputStream.close();
                System.out.println("File " + fileCacheEntry.getName() + " write to disk");
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
    }

    private class FileCacheEntry {
        private final static int state_notshared = 0;
        private final static int state_readshared = 1;
        private final static int state_writeshared = 2;
        private final static int state_ownershipchange = 3;
        private int state = state_notshared;
        private String name = "";
        private byte[] bytes = null;
        private ArrayList<String> readers = null;
        //private int suspendedDownloadInQueue = 0;
        private String owner = "";

        public FileCacheEntry() {
            readers = new ArrayList<>();
        }

        public FileCacheEntry(String name) {
            this.name = name;
            readers = new ArrayList<>();
        }
        public boolean isState_notshared() {
            return this.state == state_notshared;
        }

        public boolean isState_readshared() {
            return this.state == state_readshared;
        }

        public boolean isState_writeshared() {
            return this.state == state_writeshared;
        }

        public boolean isState_ownershipchange() {
            return this.state == state_ownershipchange;
        }


        public void setState_notshared() {
            this.state = state_notshared;
        }

        public void setState_readshared() {
            this.state = state_readshared;
        }

        public void setState_writeshared() {
            this.state = state_writeshared;
        }

        public void setState_ownershipchange() {
            this.state = state_ownershipchange;
        }

        public String getName() {
            return name;
        }

        public void setName(String name) {
            this.name = name;
        }

        public byte[] getBytes() {
            return bytes;
        }

        public void setBytes(byte[] bytes) {
            this.bytes = bytes;
        }

        public ArrayList<String> getReaders() {
            return readers;
        }

        public void addReaders(String reader) {
            readers.add(reader);
        }

        public String getOwner() {
            return owner;
        }

        public void setOwner(String owner) {
            this.owner = owner;
        }

        private String stateNameRetrieve() {
            String stateName = "";
            switch (state) {
                case 0:
                    stateName = "not_shared";
                    break;
```



```
                case 1:
                    stateName = "read_shared";
                    break;
                case 2:
                    stateName = "write_shared";
                    break;
                case 3:
                    stateName = "ownership_changed";
                    break;
            }
            return stateName;
        }

        private synchronized boolean upload(String clientName, FileContents content) {
            try {
                //check file states is ownership_change or write_shared

                if (isState_writeshared()) {
                    //If the entry state is "Write_Shared", the server changes the entry
state to "Not_Shared".
                    setState_notshared();

                } else if (isState_ownershipchange()) {
                    //If the entry state is "Ownership_Change", the server changes the entry
state to "Write_Shared"
                    setState_writeshared();
                } else {
                    return false;
                }

                //check ownership with client name
                if (owner != null && owner.equals(clientName)) {
                    //store file into entry
                    setBytes(content.get());
                } else {
                    return false;
                }
                String stateName = stateNameRetrieve();
                System.out.println("Upload request by " + clientName + ".");
                System.out.println("File: " + getName() + " FileSize: " + bytes.length +
"bytes ServerFileState_AfterUpload: " + stateName);
                System.out.println("#readers: " + readers.size() + " owner: " + (owner !=
null ? owner : "none"));
                //request invalidate
                for (String reader : readers) {
                    ((ClientInterface) Naming.lookup("rmi://" + reader + ":" + port +
"/fileclient")).invalidate();
                }
                //empty the list
                readers.clear();
                this.notify();
                return true;
            } catch (RemoteException | NotBoundException | MalformedURLException e) {
                e.printStackTrace();
                return false;
            }
        }

        private void downloadServerPrint(String clientName, String mode) {
            String stateName = stateNameRetrieve();
            System.out.println("Download request by " + clientName + ". Mode: " +
(mode.equals("r") ? "read" : "write"));
            System.out.println("File: " + getName() + "FileSize: " + bytes.length + "bytes
ServerFileState: " + stateName);
            System.out.println("#readers: " + readers.size() + " owner: " + (owner !=
null ? owner : "none"));
        }

        private synchronized FileContents download(String clientName, String mode) {
            try {
```

```
            //download as read
            if (mode.equals("r")) {
                if (isState_readshared() || isState_notshared()) {
                    //read file with read_shared/not_shared
                    readers.add(clientName);
                    setState_readshared();
                    //print server info on download
                    downloadServerPrint(clientName, mode);
                    return new FileContents(bytes);

                } else if (isState_writeshared() || isState_ownershipchange()) {
                    //read file with Ownership_Change states/write_shared states
                    readers.add(clientName);
                    //print server info on download
                    downloadServerPrint(clientName, mode);
                    return new FileContents(bytes);
                }
            } else {
                //download as write
                if (isState_readshared() || isState_notshared()) {
                    //write file with read_share states
                    owner = clientName;
                    setState_writeshared();
                    //print server info on download
                    downloadServerPrint(clientName, mode);
                    return new FileContents(bytes);

                } else if (isState_writeshared()) {
                    //write file with write_share states
                    setState_ownershipchange();
                    //print server info on download
                    downloadServerPrint(clientName, mode);
                    //request writeback
                    ((ClientInterface) Naming.lookup("rmi://" + owner + ":" + port +
"/fileclient")).writeback();
                    try {
                        System.out.println("Download suspend on " + clientName);
                        //wait for upload to notify
                        this.wait();
                        //notify the thread that wait on ownershipchagne states
                        this.notifyAll();
                        System.out.println("Download resume on " + clientName);

                    } catch (InterruptedException e) {
                        e.printStackTrace();
                    }
                    owner = clientName;
                    return new FileContents(bytes);

                } else if (isState_ownershipchange()) {
                    //write file with Ownership_Change states, resume until states change
to write_share

                    try {
                        //print server info on download
                        downloadServerPrint(clientName, mode);
                        System.out.println("Download suspend on " + clientName);
                        //wait for download to notifyAll
                        this.wait();
                        //request writeback
                        ((ClientInterface) Naming.lookup("rmi://" + owner + ":" + port +
"/fileclient")).writeback();
                        //wait for upload to notify
                        this.wait();
                        //notify the other threads that wait on ownershipchagne states
                        this.notifyAll();
                        System.out.println("Download resume on " + clientName);
                    } catch (InterruptedException e) {
                        e.printStackTrace();
                    }
                    setState_writeshared();
```

```
                        owner = clientName;
                        return new FileContents(bytes);
                    } else {
                        return null;
                    }

                }


            } catch (RemoteException | NotBoundException | MalformedURLException e) {
                e.printStackTrace();
            }
            return null;
        }
    }
}
```

Test output

(1) Compilation: steps 1-4 (3pt)

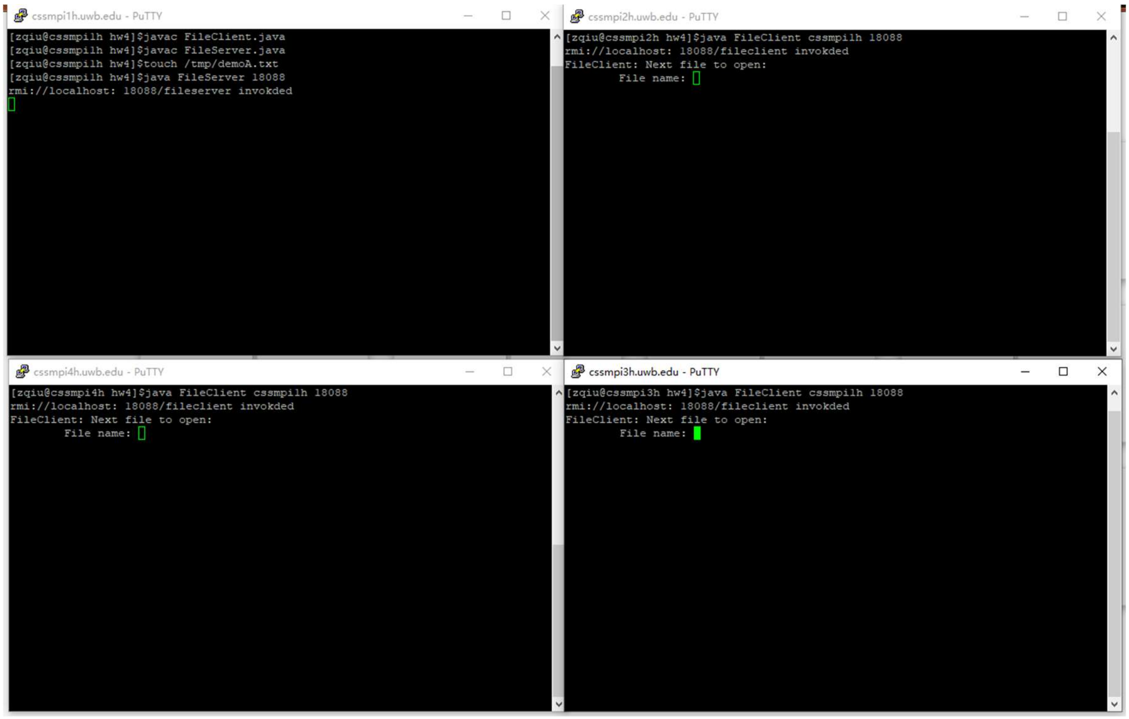
1. Open 4 windows: each logging in a different cssmpiNh machine. For example, cssmpi1h, cssmpi2h, cssmpi3h, and cssmpi4h.

2. Compile with javac.

3. Create an empty file: demoA (with a Unix command "touch demoA")

4a. Start a server on cssmpi1h.

4b. Start a client on cssmpi2h, cssmpi3h, and cssmpi4h.



```
[zqiu@cssmpi1h hw4]$javac FileClient.java
[zqiu@cssmpi1h hw4]$javac FileServer.java
[zqiu@cssmpi1h hw4]$touch /tmp/demoA.txt
[zqiu@cssmpi1h hw4]$java FileServer 18088
rmi://localhost: 18088/fileserver invokded
```

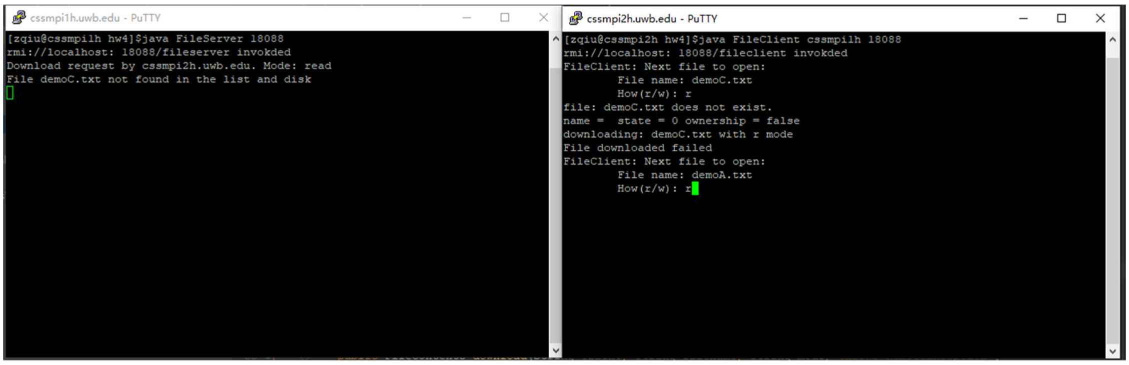
```
[zqiu@cssmpi2h hw4]$java FileClient cssmpi1h 18088
rmi://localhost: 18088/fileclient invokded
FileClient: Next file to open:
        File name:
```

```
[zqiu@cssmpi4h hw4]$java FileClient cssmpi1h 18088
rmi://localhost: 18088/fileclient invokded
FileClient: Next file to open:
        File name:
```

```
[zqiu@cssmpi3h hw4]$java FileClient cssmpi1h 18088
rmi://localhost: 18088/fileclient invokded
FileClient: Next file to open:
        File name:
```

(2) File read test: step 5 (2pt)

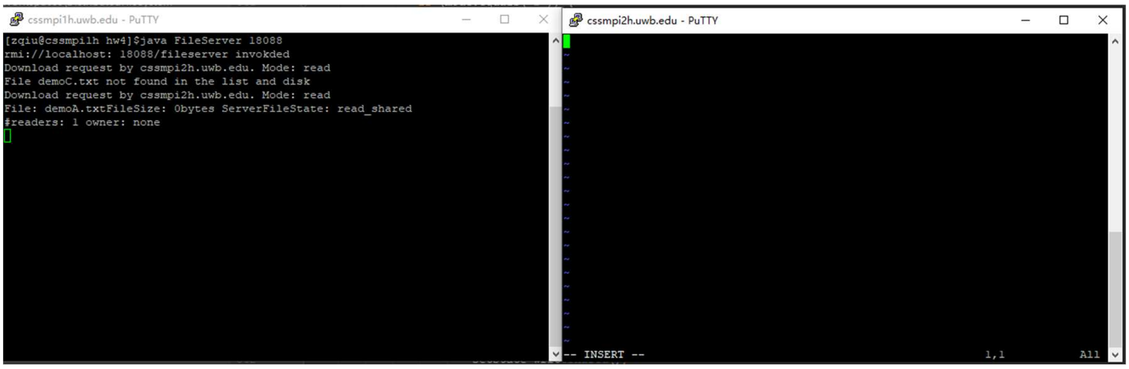
5a. Read from demoC at cssmpi2h. (a read error must be handled at the server)



```
[zqiu@cssmpi1h hw4]$java FileServer 18088
rmi://localhost: 18088/fileserver invokded
Download request by cssmpi2h.uwb.edu. Mode: read
File demoC.txt not found in the list and disk
```

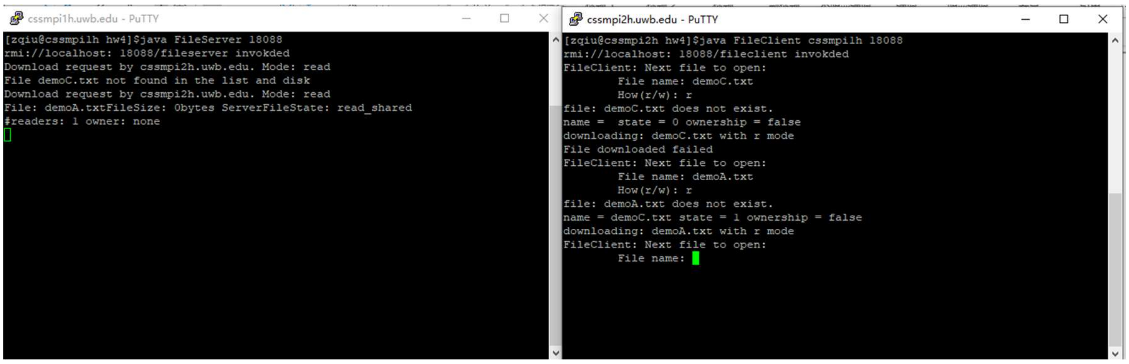
```
[zqiu@cssmpi2h hw4]$java FileClient cssmpi1h 18088
rmi://localhost: 18088/fileclient invokded
FileClient: Next file to open:
        File name: demoC.txt
        How(r/w): r
file: demoC.txt does not exist.
name =  state = 0 ownership = false
downloading: demoC.txt with r mode
File downloaded failed
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
```

5b. Read empty from demoA at cssmpi2h. (A: clt:rs, svr:rs)



```
[zqiu@cssmpi1h hw4]$java FileServer 18088
rmi://localhost: 18088/fileserver invokded
Download request by cssmpi2h.uwb.edu. Mode: read
File demoC.txt not found in the list and disk
Download request by cssmpi2h.uwb.edu. Mode: read
File: demoA.txtFileSize: 0bytes ServerFileState: read_shared
#readers: 1 owner: none
```

```

~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --                                                  1,1           All
```

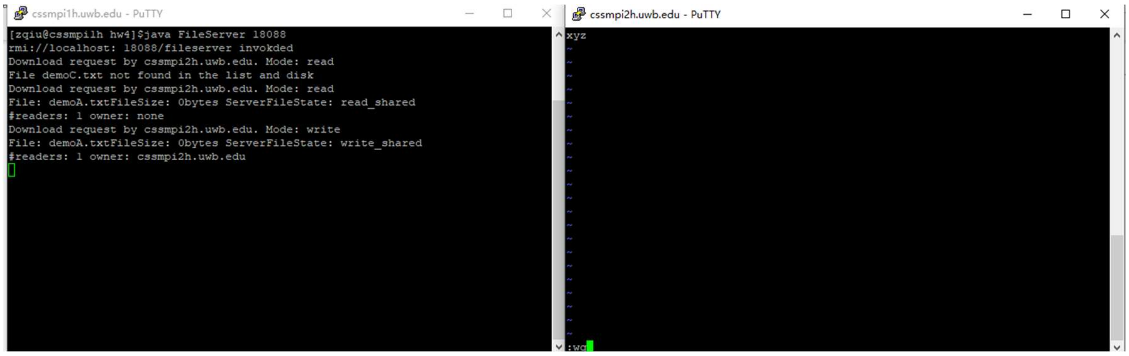


```
[zqiu@cssmpi1h hw4]$java FileServer 18088
rmi://localhost: 18088/fileserver invokded
Download request by cssmpi2h.uwb.edu. Mode: read
File demoC.txt not found in the list and disk
Download request by cssmpi2h.uwb.edu. Mode: read
File: demoA.txtFileSize: 0bytes ServerFileState: read_shared
#readers: 1 owner: none
```

```
[zqiu@cssmpi2h hw4]$java FileClient cssmpi1h 18088
rmi://localhost: 18088/fileclient invokded
FileClient: Next file to open:
        File name: demoC.txt
        How(r/w): r
file: demoC.txt does not exist.
name =  state = 0 ownership = false
downloading: demoC.txt with r mode
File downloaded failed
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt does not exist.
name = demoC.txt state = 1 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: 
```

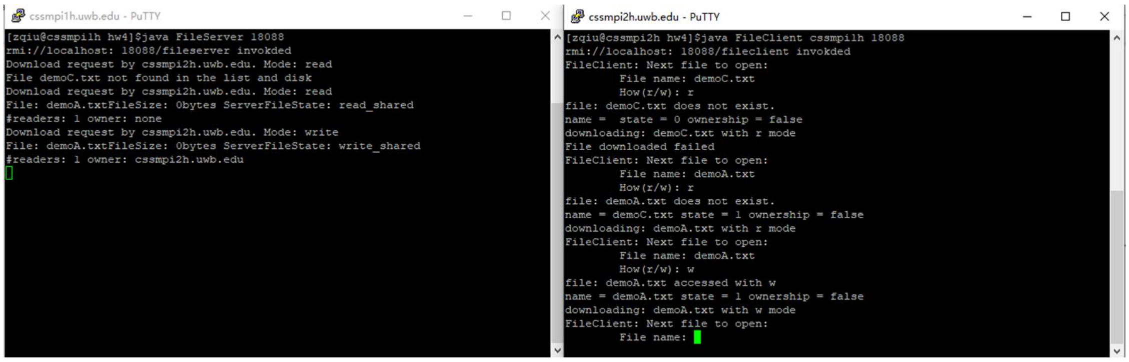
(3) File write test: steps 6 and 7 (2pt)

6. Write xyz to demoA at cssmpi2h. (A: clt:wo, svr:ws)



```
[zqiu@cssmpi1h hw4]$java FileServer 18088
rmi://localhost: 18088/fileserver invokded
Download request by cssmpi2h.uwb.edu. Mode: read
File demoC.txt not found in the list and disk
Download request by cssmpi2h.uwb.edu. Mode: read
File: demoA.txtFileSize: 0bytes ServerFileState: read_shared
#readers: 1 owner: none
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoA.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 1 owner: cssmpi2h.uwb.edu
```

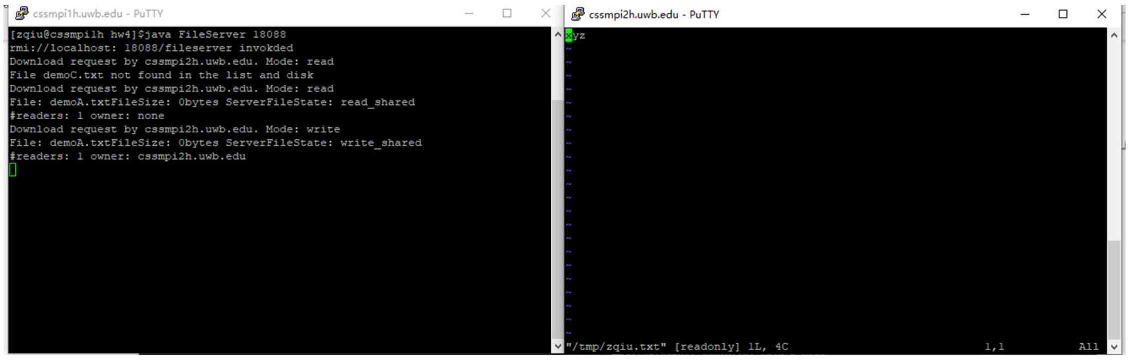
```
xyz
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:wq
```



```
[zqiu@cssmpi1h hw4]$java FileServer 18088
rmi://localhost: 18088/fileserver invokded
Download request by cssmpi2h.uwb.edu. Mode: read
File demoC.txt not found in the list and disk
Download request by cssmpi2h.uwb.edu. Mode: read
File: demoA.txtFileSize: 0bytes ServerFileState: read_shared
#readers: 1 owner: none
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoA.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 1 owner: cssmpi2h.uwb.edu
```

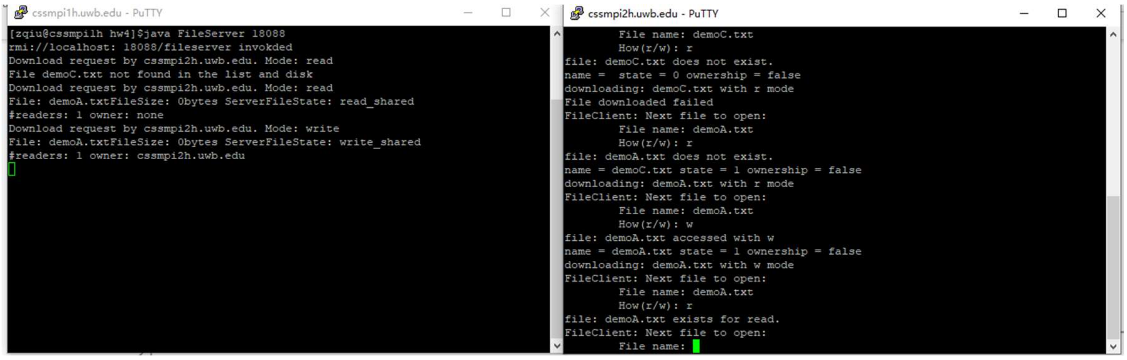
```
[zqiu@cssmpi2h hw4]$java FileClient cssmpi1h 18088
rmi://localhost: 18088/fileclient invokded
FileClient: Next file to open:
        File name: demoC.txt
        How(r/w): r
file: demoC.txt does not exist.
name =  state = 0 ownership = false
downloading: demoC.txt with r mode
File downloaded failed
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt does not exist.
name = demoC.txt state = 1 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): w
file: demoA.txt accessed with w
name = demoA.txt state = 1 ownership = false
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name:
```

7. Read xyz from demoA at cssmpi2h. (A: clt:wo, svr:ws)



```
[zqiu@cssmpi1h hw4]$java FileServer 18088
rmi://localhost: 18088/fileserver invokded
Download request by cssmpi2h.uwb.edu. Mode: read
File demoC.txt not found in the list and disk
Download request by cssmpi2h.uwb.edu. Mode: read
File: demoA.txtFileSize: 0bytes ServerFileState: read_shared
#readers: 1 owner: none
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoA.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 1 owner: cssmpi2h.uwb.edu
```

```
xyz
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"/tmp/zqiu.txt" [readonly] 1L, 4C                        1,1           All
```

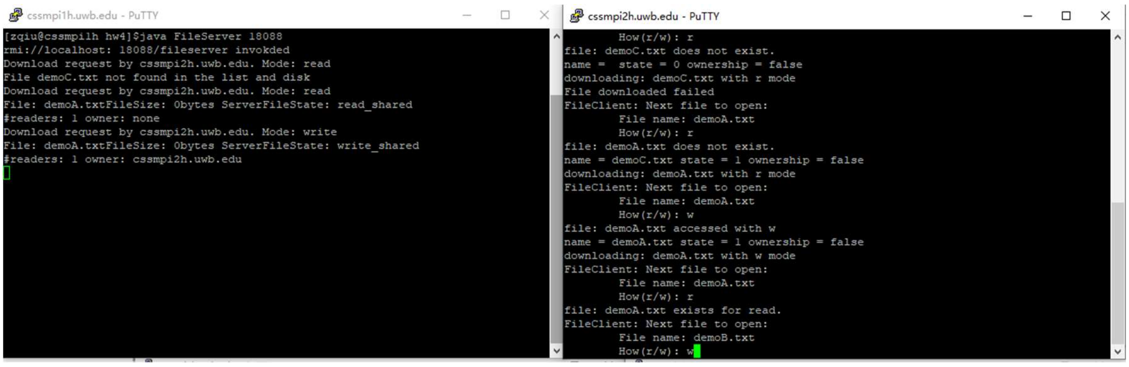


```
[zqiu@cssmpi1h hw4]$java FileServer 18088
rmi://localhost: 18088/fileserver invokded
Download request by cssmpi2h.uwb.edu. Mode: read
File demoC.txt not found in the list and disk
Download request by cssmpi2h.uwb.edu. Mode: read
File: demoA.txtFileSize: 0bytes ServerFileState: read_shared
#readers: 1 owner: none
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoA.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 1 owner: cssmpi2h.uwb.edu
```

```
        File name: demoC.txt
        How(r/w): r
file: demoC.txt does not exist.
name =  state = 0 ownership = false
downloading: demoC.txt with r mode
File downloaded failed
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt does not exist.
name = demoC.txt state = 1 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): w
file: demoA.txt accessed with w
name = demoA.txt state = 1 ownership = false
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt exists for read.
FileClient: Next file to open:
        File name:
```

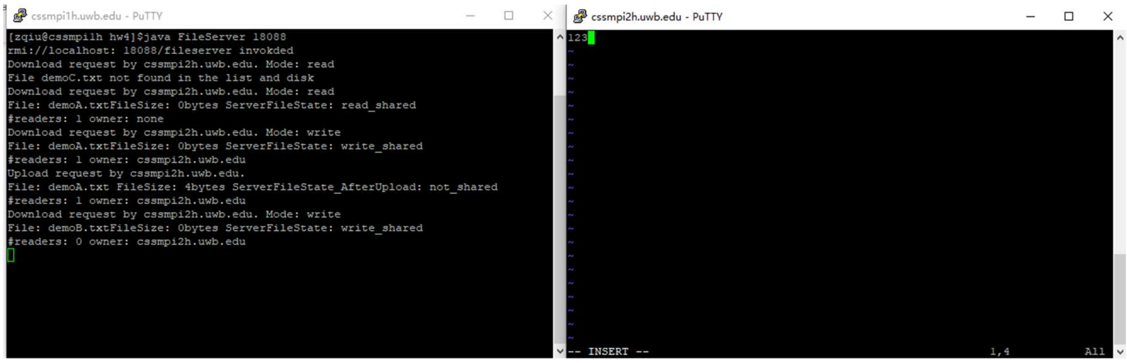
(4) File replacement test: steps 8, 9, and 10 (3pt)

8. Write 123 to demoB at cssmpi2h. (A: clt:iv, svr:ns, B: clt:wo, svr:ws)



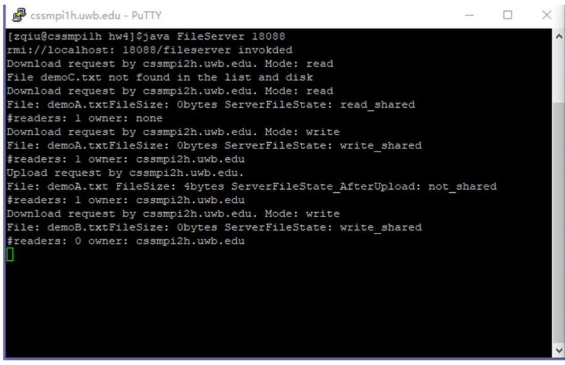
```
[zqiu@cssmpi1h hw4]$java FileServer 18088
rmi://localhost: 18088/fileserver invokded
Download request by cssmpi2h.uwb.edu. Mode: read
File demoC.txt not found in the list and disk
Download request by cssmpi2h.uwb.edu. Mode: read
File: demoA.txtFileSize: 0bytes ServerFileState: read_shared
#readers: 1 owner: none
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoA.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 1 owner: cssmpi2h.uwb.edu
```

```
        How(r/w): r
file: demoC.txt does not exist.
name =  state = 0 ownership = false
downloading: demoC.txt with r mode
File downloaded failed
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt does not exist.
name = demoC.txt state = 1 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): w
file: demoA.txt accessed with w
name = demoA.txt state = 1 ownership = false
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt exists for read.
FileClient: Next file to open:
        File name: demoB.txt
        How(r/w): w
```

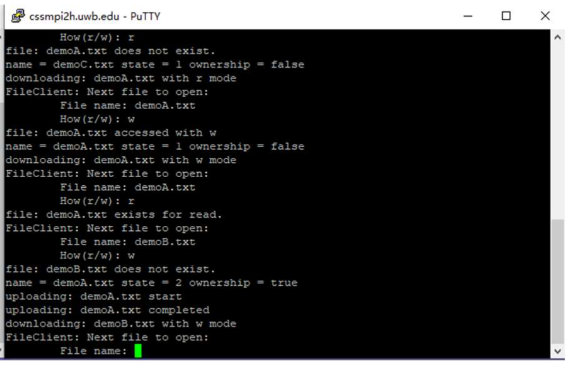


```
[zqiu@cssmpi1h hw4]$java FileServer 18088
rmi://localhost: 18088/fileserver invokded
Download request by cssmpi2h.uwb.edu. Mode: read
File demoC.txt not found in the list and disk
Download request by cssmpi2h.uwb.edu. Mode: read
File: demoA.txtFileSize: 0bytes ServerFileState: read_shared
#readers: 1 owner: none
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoA.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoA.txt FileSize: 4bytes ServerFileState_AfterUpload: not_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoB.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
```

```
123
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --                                          1,4           All
```

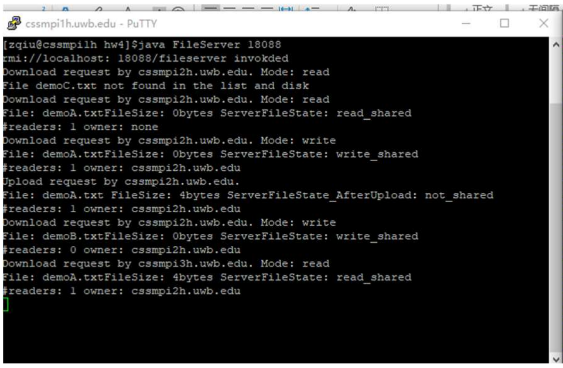


```
[zqiu@cssmpi1h hw4]$java FileServer 18088
rmi://localhost: 18088/fileserver invokded
Download request by cssmpi2h.uwb.edu. Mode: read
File demoC.txt not found in the list and disk
Download request by cssmpi2h.uwb.edu. Mode: read
File: demoA.txtFileSize: 0bytes ServerFileState: read_shared
#readers: 1 owner: none
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoA.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoA.txt FileSize: 4bytes ServerFileState_AfterUpload: not_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoB.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
```

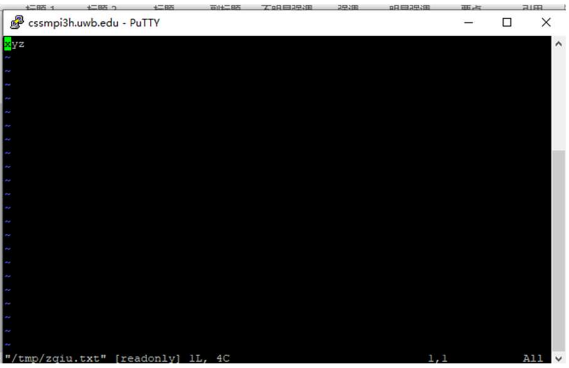


```
        How(r/w): r
file: demoA.txt does not exist.
name = demoC.txt state = 1 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): w
file: demoA.txt accessed with w
name = demoA.txt state = 1 ownership = false
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt exists for read.
FileClient: Next file to open:
        File name: demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 2 ownership = true
uploading: demoA.txt start
uploading: demoA.txt completed
downloading: demoB.txt with w mode
FileClient: Next file to open:
        File name: 
```

9. Read xyz demoA at cssmpi3h. (A: clt:rs, svr:rs)

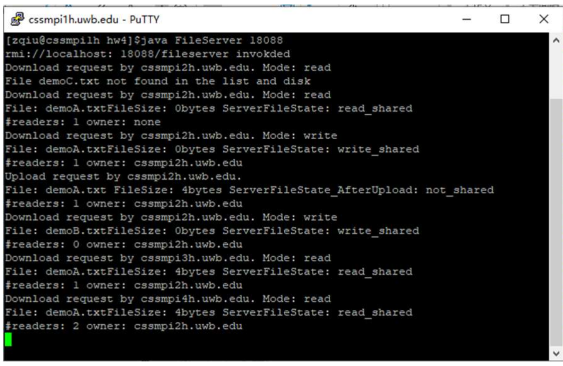


```
[zqiu@cssmpi1h hw4]$java FileServer 18088
rmi://localhost: 18088/fileserver invokded
Download request by cssmpi2h.uwb.edu. Mode: read
File demoC.txt not found in the list and disk
Download request by cssmpi2h.uwb.edu. Mode: read
File: demoA.txtFileSize: 0bytes ServerFileState: read_shared
#readers: 1 owner: none
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoA.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoA.txt FileSize: 4bytes ServerFileState_AfterUpload: not_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoB.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: read
File: demoA.txtFileSize: 4bytes ServerFileState: read_shared
#readers: 1 owner: cssmpi2h.uwb.edu
```

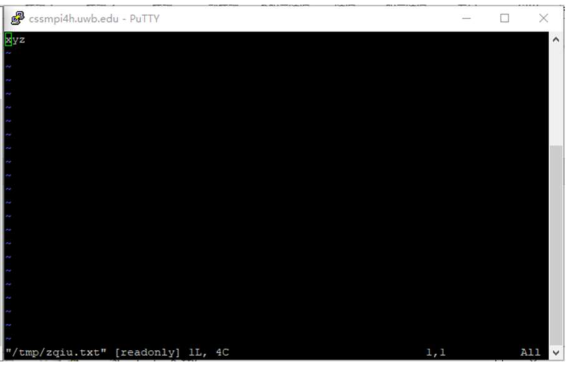


```
xyz
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"/tmp/zqiu.txt" [readonly] 1L, 4C                          1,1           All
```

10. Read xyz demoA at cssmpi4h. (A: clt:rs, svr:rs)



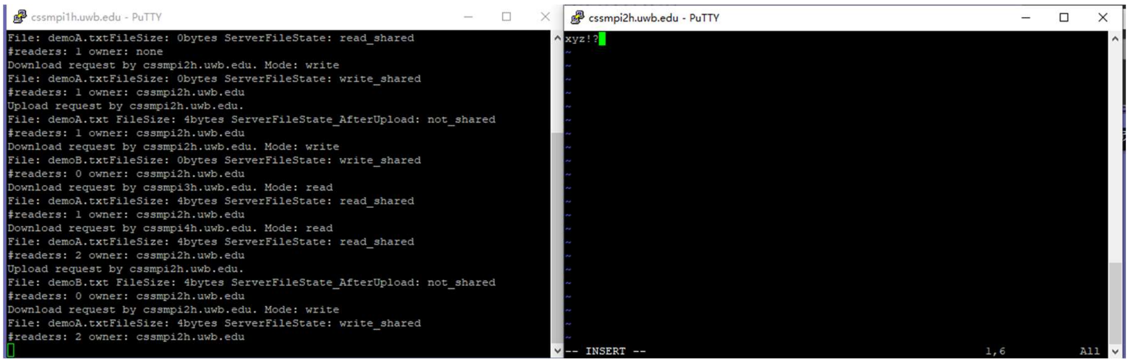
```
[zqiu@cssmpi1h hw4]$java FileServer 18088
rmi://localhost: 18088/fileserver invokded
Download request by cssmpi2h.uwb.edu. Mode: read
File demoC.txt not found in the list and disk
Download request by cssmpi2h.uwb.edu. Mode: read
File: demoA.txtFileSize: 0bytes ServerFileState: read_shared
#readers: 1 owner: none
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoA.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoA.txt FileSize: 4bytes ServerFileState_AfterUpload: not_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoB.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: read
File: demoA.txtFileSize: 4bytes ServerFileState: read_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: read
File: demoA.txtFileSize: 4bytes ServerFileState: read_shared
#readers: 2 owner: cssmpi2h.uwb.edu
```



```
xyz
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"/tmp/zqiu.txt" [readonly] 1L, 4C                          1,1           All
```

(5) File writeback test: steps 11, 13, and 15 (3pt)

11. Write xyz?! to demoA at cssmpi2h. (A: clt:wo, svr:ws, B: clt:iv, srv:ns)

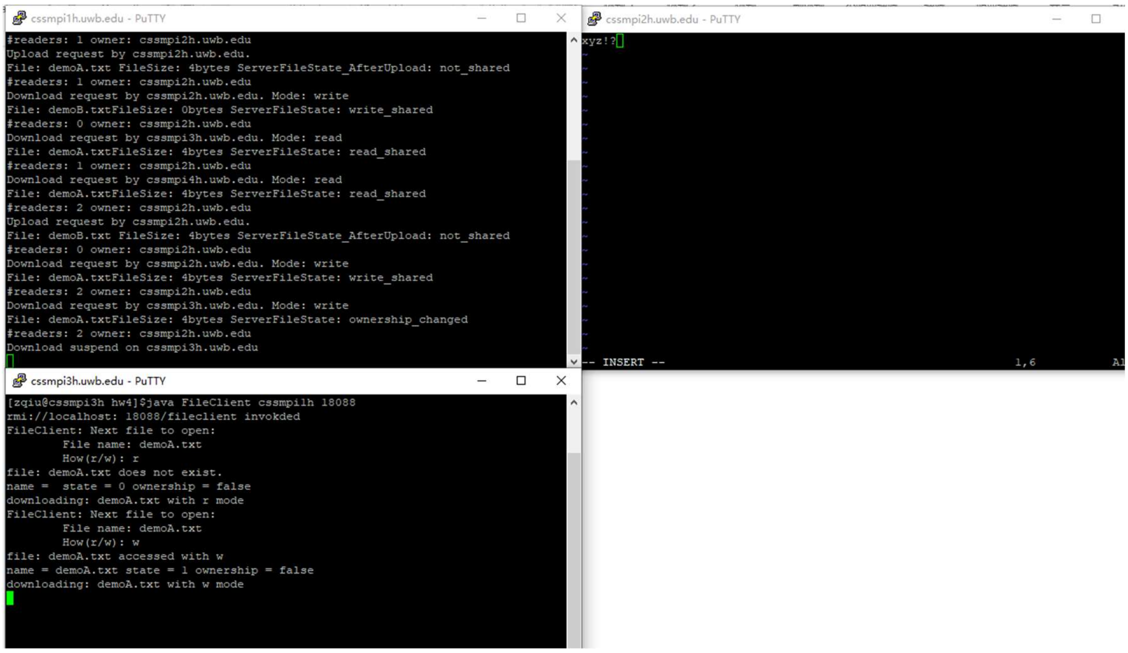


```
File: demoA.txtFileSize: 0bytes ServerFileState: read_shared
#readers: 1 owner: none
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoA.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoA.txt FileSize: 4bytes ServerFileState_AfterUpload: not_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoB.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: read
File: demoA.txtFileSize: 4bytes ServerFileState: read_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: read
File: demoA.txtFileSize: 4bytes ServerFileState: read_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoB.txt FileSize: 4bytes ServerFileState_AfterUpload: not_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoA.txtFileSize: 4bytes ServerFileState: write_shared
#readers: 2 owner: cssmpi2h.uwb.edu
```

```
xyz!?
-- INSERT --                                    1,6           All
```

12. Keep emacs open at cssmpi2h. (A: clt:wo, svr:ws, B: clt:iv, srv:ns)

13. Write to demoA at cssmpi3h. (A: clt:suspended, svr:oc)



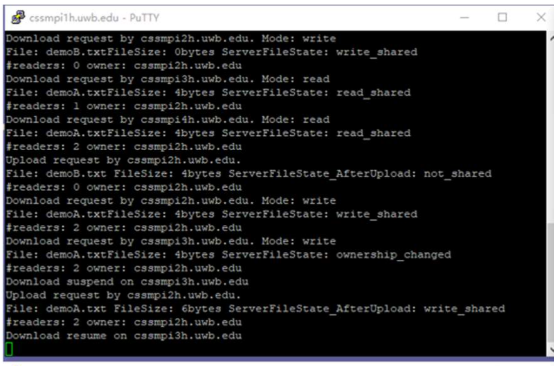
```
#readers: 1 owner: cssmpi2h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoA.txt FileSize: 4bytes ServerFileState_AfterUpload: not_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoB.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: read
File: demoA.txtFileSize: 4bytes ServerFileState: read_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: read
File: demoA.txtFileSize: 4bytes ServerFileState: read_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoB.txt FileSize: 4bytes ServerFileState_AfterUpload: not_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoA.txtFileSize: 4bytes ServerFileState: write_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: write
File: demoA.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 2 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi3h.uwb.edu
```

```
xyz!?
-- INSERT --                                    1,6           Al
```

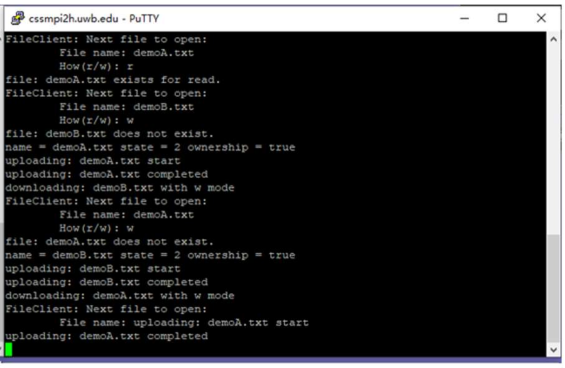
```
[zqiu@cssmpi3h hw4]$java FileClient cssmpi1h 18088
rmi://localhost: 18088/fileclient invokded
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt does not exist.
name =  state = 0 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): w
file: demoA.txt accessed with w
name = demoA.txt state = 1 ownership = false
downloading: demoA.txt with w mode
```

14. close emacs at cssmpi2h. (A: clt:rs, svr:ws)

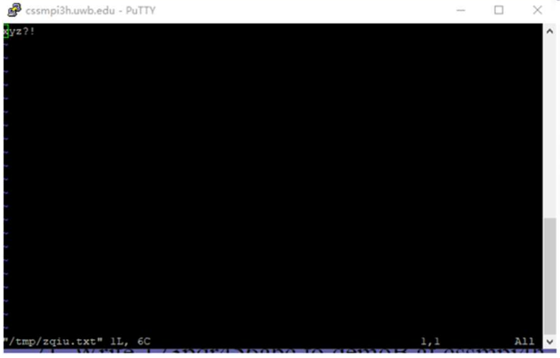
15. Write xyz?!abc to demoA at cssmpi3h (A: clt:wo, srv;ws)



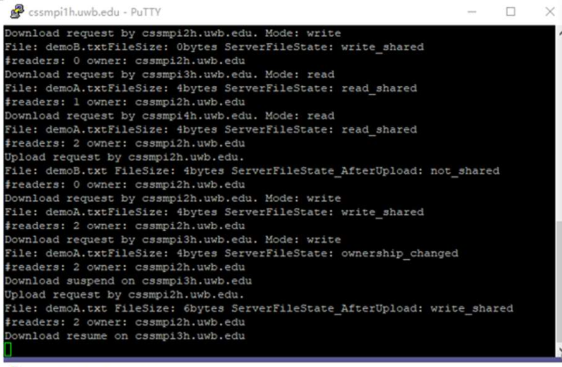
```
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoB.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: read
File: demoA.txtFileSize: 4bytes ServerFileState: read_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: read
File: demoA.txtFileSize: 4bytes ServerFileState: read_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoB.txt FileSize: 4bytes ServerFileState_AfterUpload: not_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoA.txtFileSize: 4bytes ServerFileState: write_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: write
File: demoA.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 2 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi3h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoA.txt FileSize: 6bytes ServerFileState_AfterUpload: write_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Download resume on cssmpi3h.uwb.edu
```



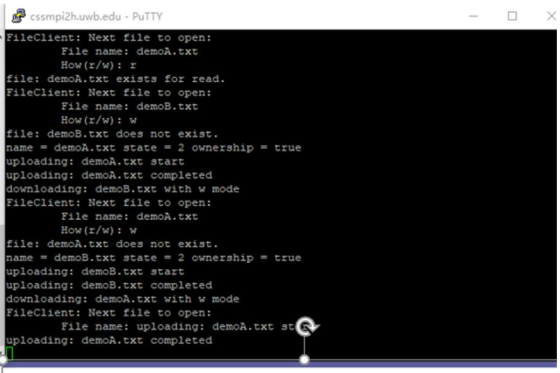
```
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt exists for read.
FileClient: Next file to open:
        File name: demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 2 ownership = true
uploading: demoA.txt start
uploading: demoA.txt completed
downloading: demoB.txt with w mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): w
file: demoA.txt does not exist.
name = demoB.txt state = 2 ownership = true
uploading: demoB.txt start
uploading: demoB.txt completed
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoA.txt start
uploading: demoA.txt completed
```



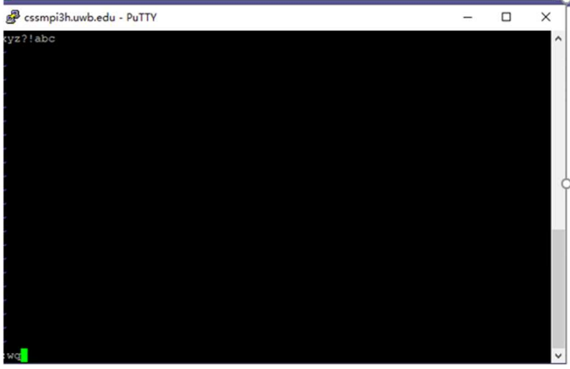
```
xyz?!









"/tmp/zqiu.txt" 1L, 6C                              1,1           All
```



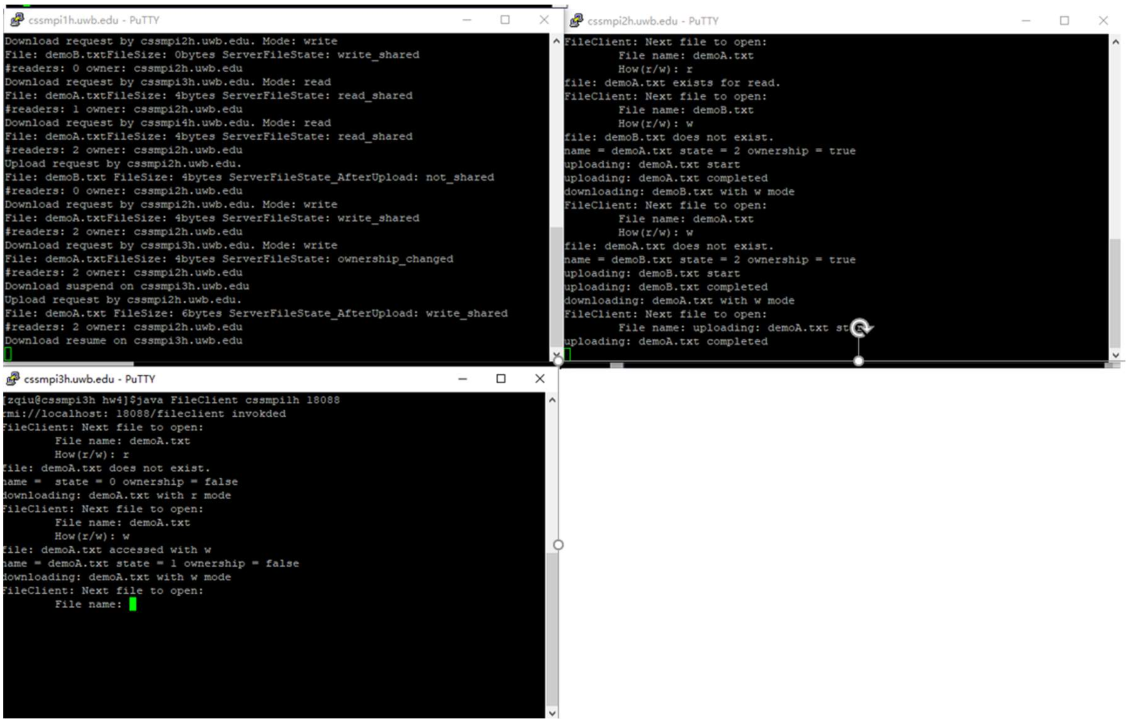
```
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoB.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: read
File: demoA.txtFileSize: 4bytes ServerFileState: read_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: read
File: demoA.txtFileSize: 4bytes ServerFileState: read_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoB.txt FileSize: 4bytes ServerFileState_AfterUpload: not_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoA.txtFileSize: 4bytes ServerFileState: write_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: write
File: demoA.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 2 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi3h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoA.txt FileSize: 6bytes ServerFileState_AfterUpload: write_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Download resume on cssmpi3h.uwb.edu
```



```
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt exists for read.
FileClient: Next file to open:
        File name: demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 2 ownership = true
uploading: demoA.txt start
uploading: demoA.txt completed
downloading: demoB.txt with w mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): w
file: demoA.txt does not exist.
name = demoB.txt state = 2 ownership = true
uploading: demoB.txt start
uploading: demoB.txt completed
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoA.txt st
uploading: demoA.txt completed
```



```
xyz?!abc









:wq
```



```
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoB.txtFileSize: 0bytes ServerFileState: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: read
File: demoA.txtFileSize: 4bytes ServerFileState: read_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: read
File: demoA.txtFileSize: 4bytes ServerFileState: read_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoB.txt FileSize: 4bytes ServerFileState_AfterUpload: not_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoA.txtFileSize: 4bytes ServerFileState: write_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: write
File: demoA.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 2 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi3h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoA.txt FileSize: 6bytes ServerFileState_AfterUpload: write_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Download resume on cssmpi3h.uwb.edu
```

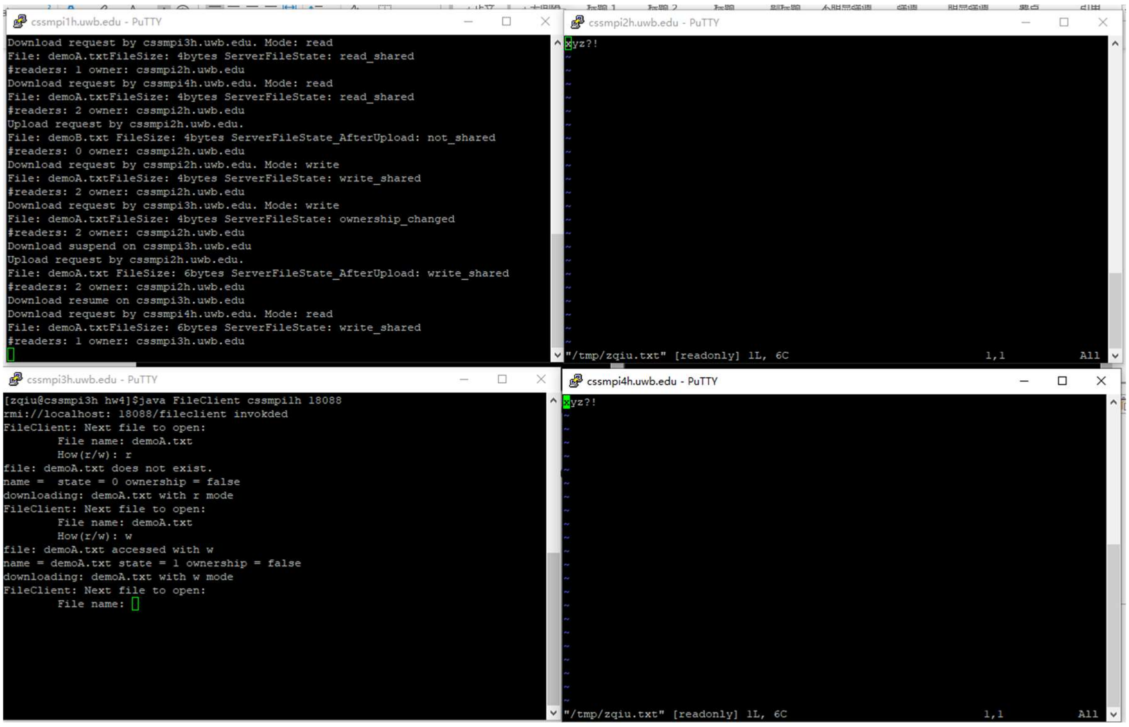
```
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt exists for read.
FileClient: Next file to open:
        File name: demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 2 ownership = true
uploading: demoA.txt start
uploading: demoA.txt completed
downloading: demoB.txt with w mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): w
file: demoA.txt does not exist.
name = demoB.txt state = 2 ownership = true
uploading: demoB.txt start
uploading: demoB.txt completed
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoA.txt st
uploading: demoA.txt completed
```

```
[zqiu@cssmpi3h hw4]$java FileClient cssmpilh 18088
rmi://localhost: 18088/fileclient invokded
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt does not exist.
name =  state = 0 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): w
file: demoA.txt accessed with w
name = demoA.txt state = 1 ownership = false
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: 
```

(6) Session semantics read test: steps 16 and 17 (2pt)

16. Read xyz?! from demoA at cssmpi2h.(A: clt:rs, svr:ws)

17. Read xyz?! from demoA at cssmpi4h.(A: clt:rs, svr:ws)

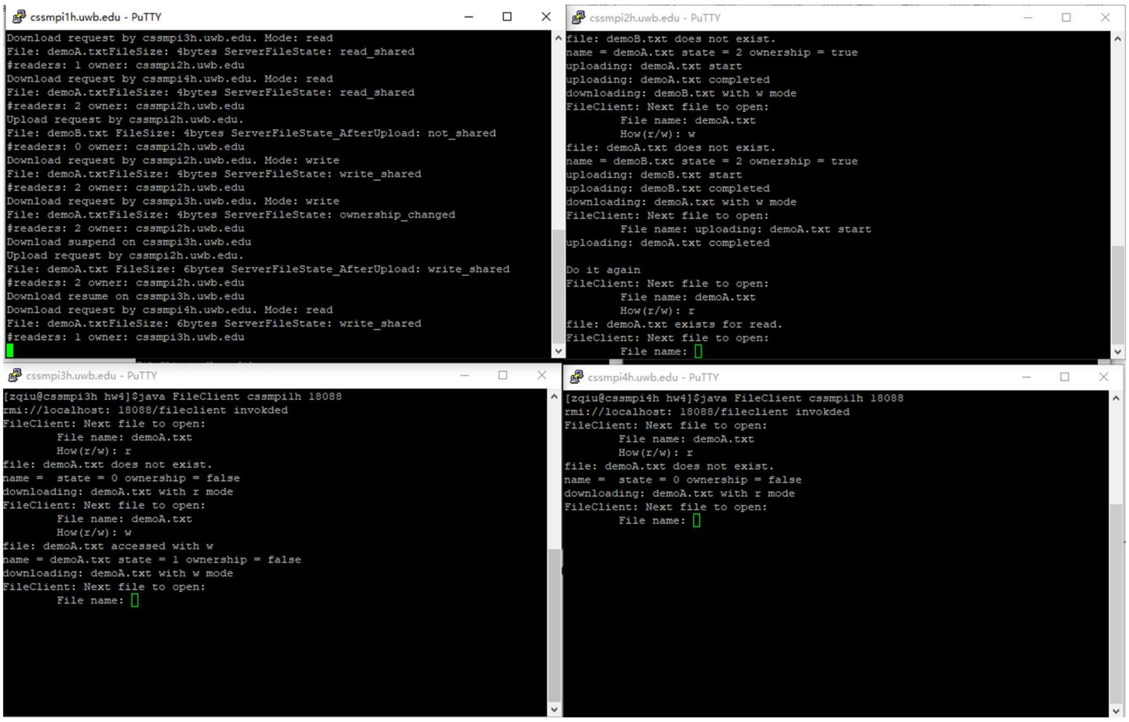


```
Download request by cssmpi3h.uwb.edu. Mode: read
File: demoA.txtFileSize: 4bytes ServerFileState: read_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: read
File: demoA.txtFileSize: 4bytes ServerFileState: read_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoB.txt FileSize: 4bytes ServerFileState_AfterUpload: not_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoA.txtFileSize: 4bytes ServerFileState: write_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: write
File: demoA.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 2 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi3h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoA.txt FileSize: 6bytes ServerFileState_AfterUpload: write_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Download resume on cssmpi3h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: read
File: demoA.txtFileSize: 6bytes ServerFileState: write_shared
#readers: 1 owner: cssmpi3h.uwb.edu
```

```
xyz?!
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"/tmp/zqiu.txt" [readonly] 1L, 6C                          1,1           All
```

```
[zqiu@cssmpi3h hw4]$java FileClient cssmpilh 18088
rmi://localhost: 18088/fileclient invokded
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt does not exist.
name =  state = 0 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): w
file: demoA.txt accessed with w
name = demoA.txt state = 1 ownership = false
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: 
```

```
xyz?!
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"/tmp/zqiu.txt" [readonly] 1L, 6C                          1,1           All
```



```
Download request by cssmpi3h.uwb.edu. Mode: read
File: demoA.txtFileSize: 4bytes ServerFileState: read_shared
#readers: 1 owner: cssmpi2h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: read
File: demoA.txtFileSize: 4bytes ServerFileState: read_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoB.txt FileSize: 4bytes ServerFileState_AfterUpload: not_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoA.txtFileSize: 4bytes ServerFileState: write_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: write
File: demoA.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 2 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi3h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoA.txt FileSize: 6bytes ServerFileState_AfterUpload: write_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Download resume on cssmpi3h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: read
File: demoA.txtFileSize: 6bytes ServerFileState: write_shared
#readers: 1 owner: cssmpi3h.uwb.edu
```

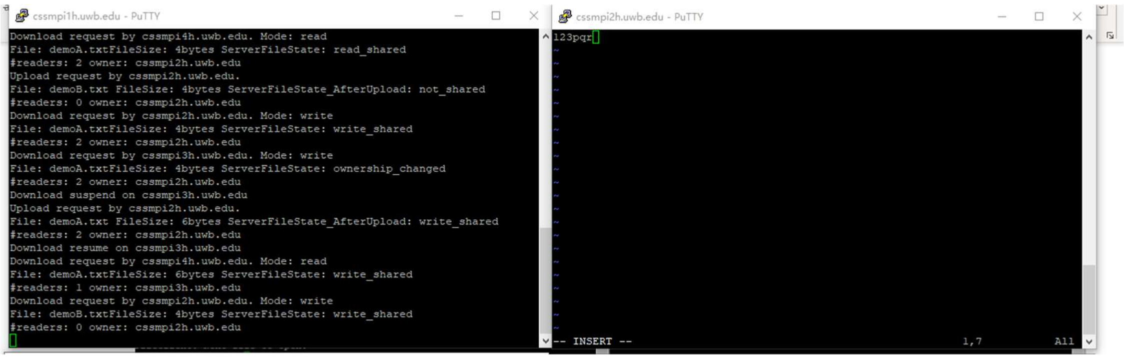
```
file: demoB.txt does not exist.
name = demoA.txt state = 2 ownership = true
uploading: demoA.txt start
uploading: demoA.txt completed
downloading: demoB.txt with w mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): w
file: demoA.txt does not exist.
name = demoB.txt state = 2 ownership = true
uploading: demoB.txt start
uploading: demoB.txt completed
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoA.txt start
uploading: demoA.txt completed

Do it again
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt exists for read.
FileClient: Next file to open:
        File name: 
```

```
[zqiu@cssmpi3h hw4]$java FileClient cssmpi1h 18088
rmi://localhost: 18088/fileclient invokded
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt does not exist.
name =  state = 0 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): w
file: demoA.txt accessed with w
name = demoA.txt state = 1 ownership = false
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: 
```

```
[zqiu@cssmpi4h hw4]$java FileClient cssmpi1h 18088
rmi://localhost: 18088/fileclient invokded
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt does not exist.
name =  state = 0 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: 
```

(7) Multiple write test: steps 18, 20, and 21 (3pt)

18. Write 123pqr to demoB at cssmpi2h.(A: clt:iv, svr:ws, B: clt:wo, svr:ws)



```
Download request by cssmpi4h.uwb.edu. Mode: read
File: demoA.txtFileSize: 4bytes ServerFileState: read_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoB.txt FileSize: 4bytes ServerFileState_AfterUpload: not_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoA.txtFileSize: 4bytes ServerFileState: write_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: write
File: demoA.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 2 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi3h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoA.txt FileSize: 6bytes ServerFileState_AfterUpload: write_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Download resume on cssmpi3h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: read
File: demoA.txtFileSize: 6bytes ServerFileState: write_shared
#readers: 1 owner: cssmpi3h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
```

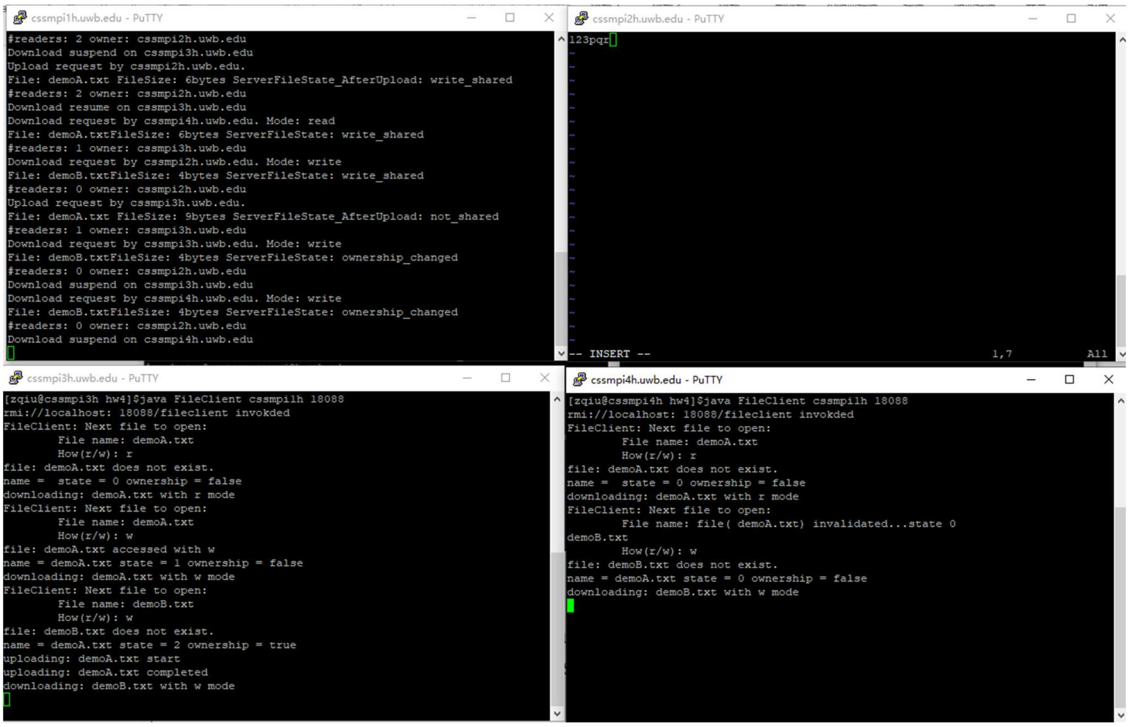
```
123pqr
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --                                        1,7           All
```

19. Keep emacs open at cssmpi2h.

20. Write 123pqr456 to demoB at cssmpi3h.(A: clt:iv, svr:rs, B: clt:suspended, svr:oc)

21. Write 123pqr456abc to demoB at cssmpi4h. (A: clt:iv, srv:ns, B: clt:suspended, svr:oc)

The screenshot of write is shown on 22&23 since the download is suspend



```
#readers: 2 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi3h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoA.txt FileSize: 6bytes ServerFileState_AfterUpload: write_shared
#readers: 2 owner: cssmpi2h.uwb.edu
Download resume on cssmpi3h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: read
File: demoA.txtFileSize: 6bytes ServerFileState: write_shared
#readers: 1 owner: cssmpi3h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Upload request by cssmpi3h.uwb.edu.
File: demoA.txt FileSize: 9bytes ServerFileState_AfterUpload: not_shared
#readers: 1 owner: cssmpi3h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 0 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi3h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 0 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi4h.uwb.edu
```

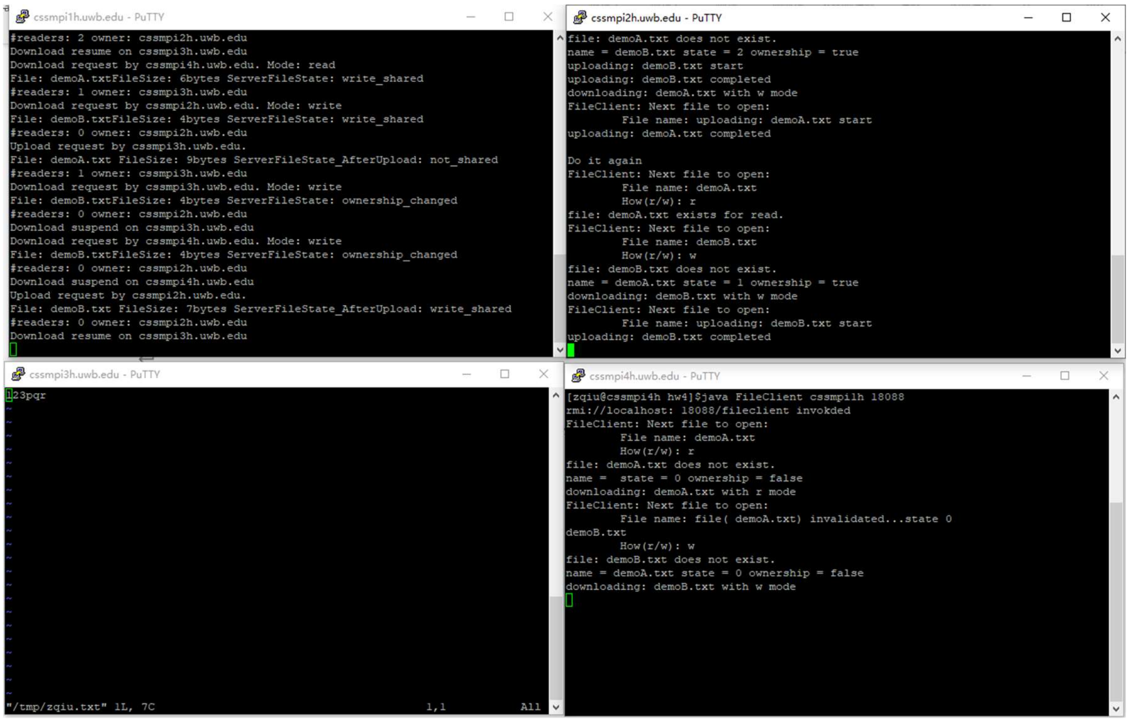
```
123pqr
~
~
~
~
-- INSERT --                                                1,7          All
```

```
[zqiu@cssmpi3h hw4]$java FileClient cssmpi1h 18088
rmi://localhost: 18088/fileclient invokded
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt does not exist.
name =  state = 0 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): w
file: demoA.txt accessed with w
name = demoA.txt state = 1 ownership = false
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 2 ownership = true
uploading: demoA.txt start
uploading: demoA.txt completed
downloading: demoB.txt with w mode
```

```
[zqiu@cssmpi4h hw4]$java FileClient cssmpi1h 18088
rmi://localhost: 18088/fileclient invokded
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt does not exist.
name =  state = 0 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: file( demoA.txt) invalidated...state 0
demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 0 ownership = false
downloading: demoB.txt with w mode
```

(8) Quiet termination: steps 22, 23, 24, 25, and 26 (2pts)

22. Close emacs at cssmpi2h. (A: clt:iv, svr:ws)

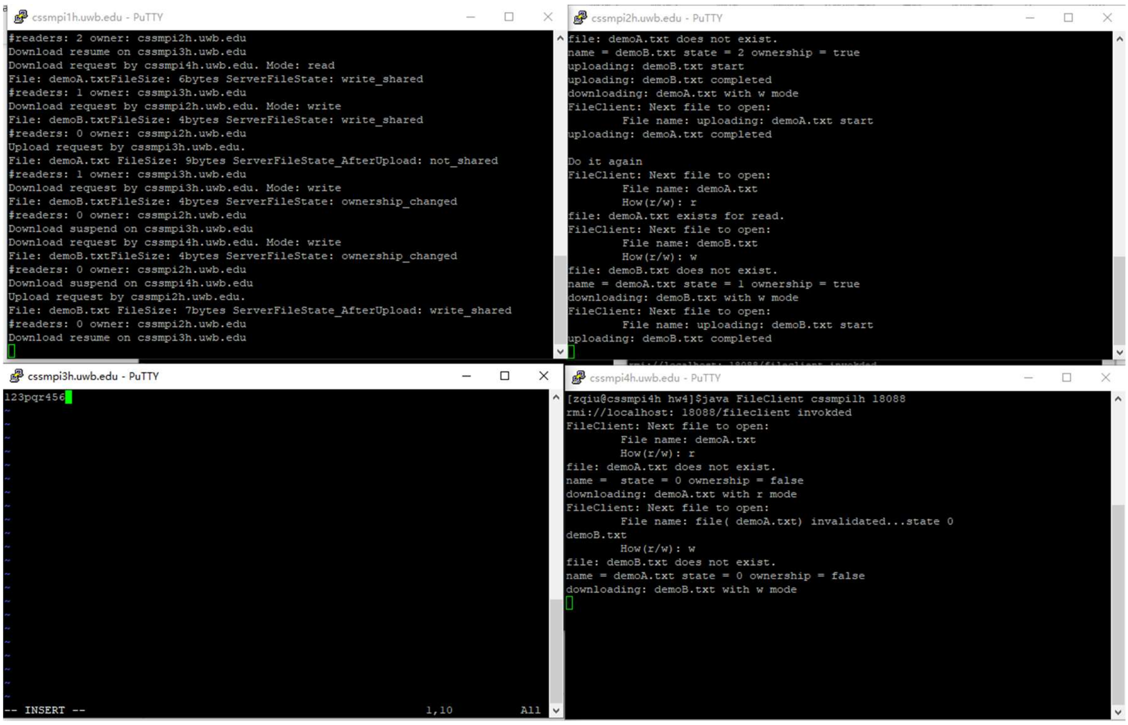


```
#readers: 2 owner: cssmpi2h.uwb.edu
Download resume on cssmpi3h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: read
File: demoA.txtFileSize: 6bytes ServerFileState: write_shared
#readers: 1 owner: cssmpi3h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Upload request by cssmpi3h.uwb.edu.
File: demoA.txt FileSize: 9bytes ServerFileState_AfterUpload: not_shared
#readers: 1 owner: cssmpi3h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 0 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi3h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 0 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi4h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoB.txt FileSize: 7bytes ServerFileState_AfterUpload: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download resume on cssmpi3h.uwb.edu
```

```
file: demoA.txt does not exist.
name = demoB.txt state = 2 ownership = true
uploading: demoB.txt start
uploading: demoB.txt completed
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoA.txt start
uploading: demoA.txt completed

Do it again
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt exists for read.
FileClient: Next file to open:
        File name: demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 1 ownership = true
downloading: demoB.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoB.txt start
uploading: demoB.txt completed
```

```
123pqr
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"/tmp/zqiu.txt" 1L, 7C                                1,1           All
```

```
[zqiu@cssmpi4h hw4]$java FileClient cssmpilh 18088
rmi://localhost: 18088/fileclient invokded
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt does not exist.
name =  state = 0 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: file( demoA.txt) invalidated...state 0
demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 0 ownership = false
downloading: demoB.txt with w mode
```



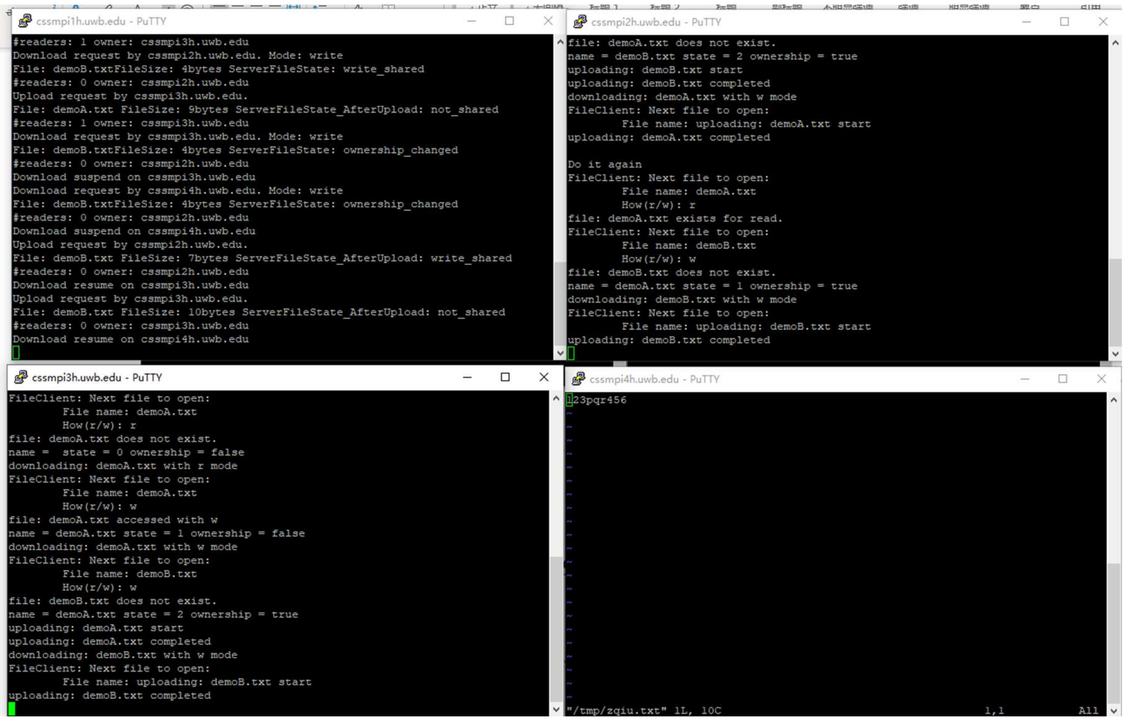
```
#readers: 2 owner: cssmpi2h.uwb.edu
Download resume on cssmpi3h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: read
File: demoA.txtFileSize: 6bytes ServerFileState: write_shared
#readers: 1 owner: cssmpi3h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Upload request by cssmpi3h.uwb.edu.
File: demoA.txt FileSize: 9bytes ServerFileState_AfterUpload: not_shared
#readers: 1 owner: cssmpi3h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 0 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi3h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 0 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi4h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoB.txt FileSize: 7bytes ServerFileState_AfterUpload: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download resume on cssmpi3h.uwb.edu
```

```
file: demoA.txt does not exist.
name = demoB.txt state = 2 ownership = true
uploading: demoB.txt start
uploading: demoB.txt completed
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoA.txt start
uploading: demoA.txt completed

Do it again
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt exists for read.
FileClient: Next file to open:
        File name: demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 1 ownership = true
downloading: demoB.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoB.txt start
uploading: demoB.txt completed
```

```
123pqr456
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --                                          1,10          All
```

```
[zqiu@cssmpi4h hw4]$java FileClient cssmpilh 18088
rmi://localhost: 18088/fileclient invokded
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt does not exist.
name =  state = 0 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: file( demoA.txt) invalidated...state 0
demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 0 ownership = false
downloading: demoB.txt with w mode
```

23. Close emacs at cssmpi3h. (A: clt:iv, svr:ws)

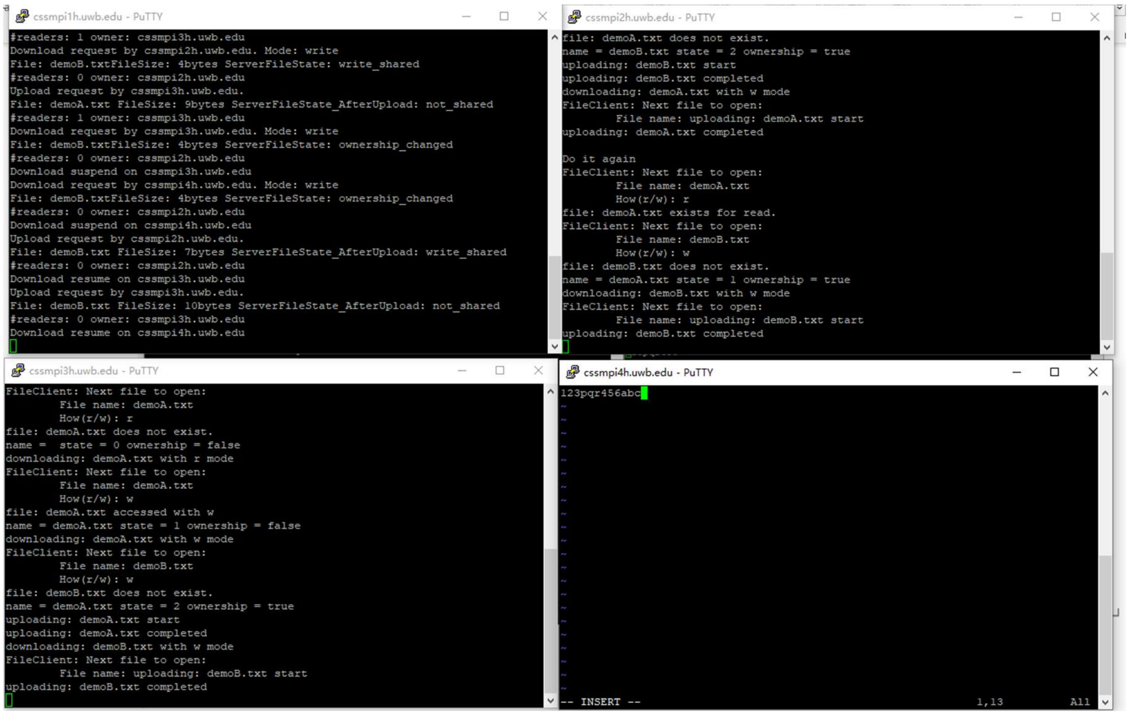


```
#readers: 1 owner: cssmpi3h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Upload request by cssmpi3h.uwb.edu.
File: demoA.txt FileSize: 9bytes ServerFileState_AfterUpload: not_shared
#readers: 1 owner: cssmpi3h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 0 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi3h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 0 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi4h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoB.txt FileSize: 7bytes ServerFileState_AfterUpload: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download resume on cssmpi3h.uwb.edu
Upload request by cssmpi3h.uwb.edu.
File: demoB.txt FileSize: 10bytes ServerFileState_AfterUpload: not_shared
#readers: 0 owner: cssmpi3h.uwb.edu
Download resume on cssmpi4h.uwb.edu
```

```
file: demoA.txt does not exist.
name = demoB.txt state = 2 ownership = true
uploading: demoB.txt start
uploading: demoB.txt completed
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoA.txt start
uploading: demoA.txt completed

Do it again
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt exists for read.
FileClient: Next file to open:
        File name: demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 1 ownership = true
downloading: demoB.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoB.txt start
uploading: demoB.txt completed
```

```
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt does not exist.
name =  state = 0 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): w
file: demoA.txt accessed with w
name = demoA.txt state = 1 ownership = false
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 2 ownership = true
uploading: demoA.txt start
uploading: demoA.txt completed
downloading: demoB.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoB.txt start
uploading: demoB.txt completed
```

```
123pqr456
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"/tmp/zqiu.txt" 1L, 10C                                  1,1           All
```



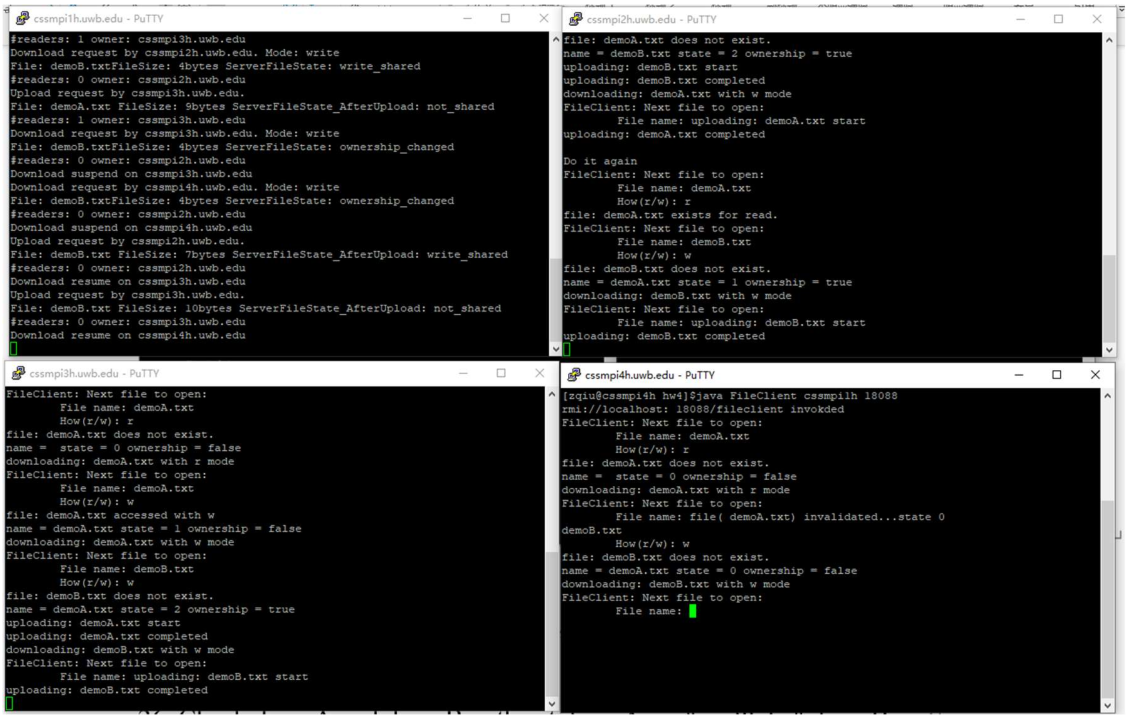
```
#readers: 1 owner: cssmpi3h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Upload request by cssmpi3h.uwb.edu.
File: demoA.txt FileSize: 9bytes ServerFileState_AfterUpload: not_shared
#readers: 1 owner: cssmpi3h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 0 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi3h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 0 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi4h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoB.txt FileSize: 7bytes ServerFileState_AfterUpload: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download resume on cssmpi3h.uwb.edu
Upload request by cssmpi3h.uwb.edu.
File: demoB.txt FileSize: 10bytes ServerFileState_AfterUpload: not_shared
#readers: 0 owner: cssmpi3h.uwb.edu
Download resume on cssmpi4h.uwb.edu
```

```
file: demoA.txt does not exist.
name = demoB.txt state = 2 ownership = true
uploading: demoB.txt start
uploading: demoB.txt completed
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoA.txt start
uploading: demoA.txt completed

Do it again
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt exists for read.
FileClient: Next file to open:
        File name: demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 1 ownership = true
downloading: demoB.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoB.txt start
uploading: demoB.txt completed
```

```
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt does not exist.
name =  state = 0 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): w
file: demoA.txt accessed with w
name = demoA.txt state = 1 ownership = false
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 2 ownership = true
uploading: demoA.txt start
uploading: demoA.txt completed
downloading: demoB.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoB.txt start
uploading: demoB.txt completed
```

```
123pqr456abc
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
-- INSERT --                                             1,13          All
```

24. Close emacs at cssmpi4h. (A: clt:wo, svr:ws)



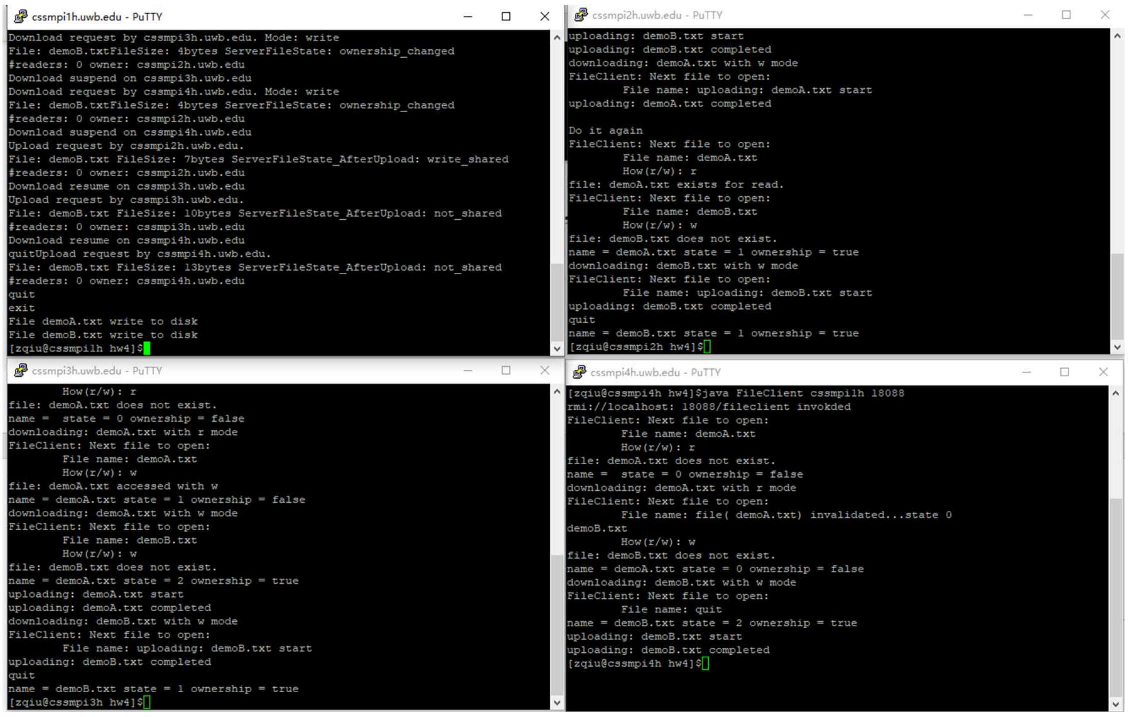
```
#readers: 1 owner: cssmpi3h.uwb.edu
Download request by cssmpi2h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Upload request by cssmpi3h.uwb.edu.
File: demoA.txt FileSize: 9bytes ServerFileState_AfterUpload: not_shared
#readers: 1 owner: cssmpi3h.uwb.edu
Download request by cssmpi3h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 0 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi3h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 0 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi4h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoB.txt FileSize: 7bytes ServerFileState_AfterUpload: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download resume on cssmpi3h.uwb.edu
Upload request by cssmpi3h.uwb.edu.
File: demoB.txt FileSize: 10bytes ServerFileState_AfterUpload: not_shared
#readers: 0 owner: cssmpi3h.uwb.edu
Download resume on cssmpi4h.uwb.edu
```

```
file: demoA.txt does not exist.
name = demoB.txt state = 2 ownership = true
uploading: demoB.txt start
uploading: demoB.txt completed
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoA.txt start
uploading: demoA.txt completed

Do it again
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt exists for read.
FileClient: Next file to open:
        File name: demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 1 ownership = true
downloading: demoB.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoB.txt start
uploading: demoB.txt completed
```

```
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt does not exist.
name =  state = 0 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): w
file: demoA.txt accessed with w
name = demoA.txt state = 1 ownership = false
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 2 ownership = true
uploading: demoA.txt start
uploading: demoA.txt completed
downloading: demoB.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoB.txt start
uploading: demoB.txt completed
```

```
[zqiu@cssmpi4h hw4]$java FileClient cssmpi1h 18088
rmi://localhost: 18088/fileclient invokded
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt does not exist.
name =  state = 0 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: file( demoA.txt) invalidated...state 0
demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 0 ownership = false
downloading: demoB.txt with w mode
FileClient: Next file to open:
        File name: 
```

25. quit cssmpi1h, cssmpi2h, cssmpi3h, and cssmpi4h.



```
Download request by cssmpi3h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 0 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi3h.uwb.edu
Download request by cssmpi4h.uwb.edu. Mode: write
File: demoB.txtFileSize: 4bytes ServerFileState: ownership_changed
#readers: 0 owner: cssmpi2h.uwb.edu
Download suspend on cssmpi4h.uwb.edu
Upload request by cssmpi2h.uwb.edu.
File: demoB.txt FileSize: 7bytes ServerFileState_AfterUpload: write_shared
#readers: 0 owner: cssmpi2h.uwb.edu
Download resume on cssmpi3h.uwb.edu
Upload request by cssmpi3h.uwb.edu.
File: demoB.txt FileSize: 10bytes ServerFileState_AfterUpload: not_shared
#readers: 0 owner: cssmpi3h.uwb.edu
Download resume on cssmpi4h.uwb.edu
quitUpload request by cssmpi4h.uwb.edu.
File: demoB.txt FileSize: 13bytes ServerFileState_AfterUpload: not_shared
#readers: 0 owner: cssmpi4h.uwb.edu
quit
exit
File demoA.txt write to disk
File demoB.txt write to disk
[zqiu@cssmpi1h hw4]$
```

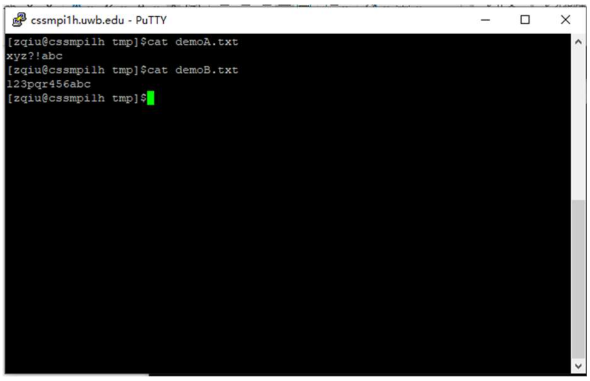
```
uploading: demoB.txt start
uploading: demoB.txt completed
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoA.txt start
uploading: demoA.txt completed

Do it again
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt exists for read.
FileClient: Next file to open:
        File name: demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 1 ownership = true
downloading: demoB.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoB.txt start
uploading: demoB.txt completed
quit
name = demoB.txt state = 1 ownership = true
[zqiu@cssmpi2h hw4]$
```

```
        How(r/w): r
file: demoA.txt does not exist.
name =  state = 0 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): w
file: demoA.txt accessed with w
name = demoA.txt state = 1 ownership = false
downloading: demoA.txt with w mode
FileClient: Next file to open:
        File name: demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 2 ownership = true
uploading: demoA.txt start
uploading: demoA.txt completed
downloading: demoB.txt with w mode
FileClient: Next file to open:
        File name: uploading: demoB.txt start
uploading: demoB.txt completed
quit
name = demoB.txt state = 1 ownership = true
[zqiu@cssmpi3h hw4]$
```

```
[zqiu@cssmpi4h hw4]$java FileClient cssmpi1h 18088
rmi://localhost: 18088/fileclient invokded
FileClient: Next file to open:
        File name: demoA.txt
        How(r/w): r
file: demoA.txt does not exist.
name =  state = 0 ownership = false
downloading: demoA.txt with r mode
FileClient: Next file to open:
        File name: file( demoA.txt) invalidated...state 0
demoB.txt
        How(r/w): w
file: demoB.txt does not exist.
name = demoA.txt state = 0 ownership = false
downloading: demoB.txt with w mode
FileClient: Next file to open:
        File name: quit
name = demoB.txt state = 2 ownership = true
uploading: demoB.txt start
uploading: demoB.txt completed
[zqiu@cssmpi4h hw4]$
```

(FileServer only recognizes “exit” as termination demand during the test, it has gotten fixed in the code)

26. Check demoA and demoB with cat demoA == "xyz?!abc" demoB =="123pqr456abc"



```
[zqiu@cssmpilh tmp]$cat demoA.txt
xyz?!abc
[zqiu@cssmpilh tmp]$cat demoB.txt
123pqr456abc
[zqiu@cssmpilh tmp]$
```

Discussion

In the discussion, I would like to discuss about two optional algorithm that I came up with during the design of this program. They might don't really have an improvement, but they are various design of implementation.

Functional improvements

- Ask writeback() right after each download for write() rather than request writeback() on busy write request (a download request for write when state is write_shared/ ownership_changed)

The goal of this design is to request the modified file update as soon as possible to avoid the case that the client crash may lost the file modification on cache. But it also has a weakness of this design, if the client frequently changes the same in one session login, it needs to upload on each modification, so it might cause high traffic issue. Moreover, a change also needs to be implemented on ownership algorithm, because in the current design the ownership of this file would be lost after upload and that would cause multiple downloads on each write request. My solution addressed to this issue is to have an ownershipInvalidation() RMI call on client side, and the server will call it when there's another client request the download for write.

Performance improvement

- Use inQueue index instead of wait() & notify() for suspending

The goal of this design is to let the thread resume followed the sequence of first-in first-out.

This algorithm can be implemented by having a currentSeq variable which is used to record the largest current queue index, and a readyInQueueSeq variable which is used to notify which suspend thread in queue is now ready to be resumed, when a thread starts FileCacheEntry's download(), the currentSeq will be increased 1 if it realize it will be suspend at this time, and then it record the currentSeq as an its own sequence number in the local function variable ownSeq, after that it runs into a dead loop which only get break when (ownSeq == readyInQueueSeq). The readyInQueueSeq will be increased 1 on the execution of upload(). With this design, the threads which suspend on ownership_changed status would not be random resumed but follow the queue sequence.