<<Java Class>>
TFTPServer
(default package)

- verboseMode: boolean = false
- serverDirectory: String = "C:\\Users\\Sherry Wang\\Desktop"
- TFTPServer()
- receiveAndSendTFTP():void
- configServer():void
- main(String[]):void

<<Java Class>>
TFTPFunctions
(default package)

- TIMEOUT: int = 4000
- MAX_TIMEOUT: int = 5
- receivePacket: DatagramPacket
- sendPacket: DatagramPacket
- ERROR_CODE_FILE_NOT_FOUND: int = 1
- ERROR_CODE_ACCESS_VIOLATION: int = 2
- ERROR_CODE_DISK_FULL: int = 3
- ERROR_CODE_ILLEGAL_TFTP_OPERATION: int = 4
- ERROR_CODE_UNKNOWN_TID: int = 5
- TFTPFunctions()
- readFileIntoBlocks(String,DatagramSocket,InetAddress,int):ArrayList<byte[]>
- createErrorPacket(InetAddress,int,int,String):DatagramPacket
- blockNumBytes(int):byte[]
- verboseMode(InetAddress,int,int,byte[]):void
- sendPacketFromSocket(DatagramSocket,DatagramPacket):void
- receivePacketFromSocket(DatagramSocket,DatagramPacket):void
- receiveFiles(String,int,String,DatagramSocket,boolean,boolean,boolean,int):void
- transferFiles(DatagramSocket,DatagramPacket,String,String,int,ArrayList<Integer>,boolean):void

<<Java Class>>
TFTPClientConnectionThread
(default package)

- receiveSocket: DatagramSocket
- receivePacket: DatagramPacket
- connectionPort: int
- verboseMode: boolean = false
- readResp: byte[] = { 0, 3, 0, 1 }
- writeResp: byte[] = { 0, 4, 0, 0 }
- fileName: String
- doneProcessingRequest: boolean = true
- establishedCommunications: ArrayList<String> = new ArrayList<String>()
- processedACKBlocks: ArrayList<Integer> = new ArrayList<>()
- serverDirectory: String
- TFTPClientConnectionThread(boolean,String)
- closeSocket():void
- checkRequestForError(int,byte[]):void
- run():void

<<Java Class>>
TFTPClient
(default package)

- sendPacket: DatagramPacket
- receivePacket: DatagramPacket
- sendReceiveSocket: DatagramSocket
- connectionPort: int
- verboseMode: boolean = false
- ipAddress: String = "192.168.0.21"
- clientDirectory: String = "C:\\Users\\Sherry Wang\\Documents\\GitHub\\SYSC3303_Project\\src"
- finishedRequest: boolean = false
- changeMode: boolean = true
- running: boolean = true
- processedACKBlocks: ArrayList<Integer> = new ArrayList<>()
- TFTPClient()
- sendAndReceive():void
- configClient():void
- main(String[]):void

<<Java Class>>
TFTPSim
(default package)

- sendPacket: DatagramPacket
- receivePacket: DatagramPacket
- receiveSocket: DatagramSocket
- sendSocket: DatagramSocket
- sendReceiveSocket: DatagramSocket
- delayTime: long = 0
- packetNumber: int = 0
- verboseMode: boolean = true
- clientPort: int
- serverPort: int = 69
- clientAddress: InetAddress
- TFTPSim()
- delayMode():void
- duplicateMode(byte[],int,InetAddress,int,String,String,int,Type):void
- lossMode(DatagramSocket,Type,boolean):void
- getType(byte[]):Type
- passOnTFTP():void
- checkIfLastDataPacket(byte[],DatagramSocket,DatagramSocket,InetAddress,int):boolean
- configSim():void
- createUnknownTIDTestThread(DatagramPacket):void
- error4Packet(byte[],int,InetAddress,int):DatagramPacket
- main(String[]):void

<<Java Class>>
TFTPsendThread
(default package)

- sendReceiveSocket: DatagramSocket
- receivePacket: DatagramPacket
- sendPacket: DatagramPacket
- verboseMode: boolean = false
- TFTPsendThread(Request,DatagramPacket,boolean)
- run():void
- send(byte[]):void

<<Java Enumeration>>
Mode
(default package)

- NORMAL: Mode
- TEST: Mode
- Mode()

<<Java Enumeration>>
RequestType
(default package)

- READ: RequestType
- WRITE: RequestType
- RequestType()

<<Java Enumeration>>
Mode
(default package)

- NORMAL: Mode
- LOSS: Mode
- DELAY: Mode
- DUPLICATE: Mode
- ERROR4: Mode
- ERROR5: Mode
- Mode()

<<Java Class>>
unknownTIDTestThread
(default package)

- packet: DatagramPacket
- unknownTIDTestThread(DatagramPacket)
- run():void

<<Java Enumeration>>
Err4Mode
(default package)

- OPCODE: Err4Mode
- BLOCKNUM: Err4Mode
- FILENAME: Err4Mode
- MODE: Err4Mode
- Err4Mode()

<<Java Enumeration>>
Type
(default package)

- RRQ: Type
- WRQ: Type
- REQ: Type
- DATA: Type
- ACK: Type
- Type()

<<Java Enumeration>>
Request
(default package)

- READ: Request
- WRITE: Request
- ERROR: Request
- Request()

-run 0..1
-request 0..1
-mode 0..1
-err4Mode 0..1
-packetType 0..1
-request 0..1

*Key note: From Iteration 1-4 we had encapsulated the code to be able to handle a scenario where the client and the server can reside on different computers.