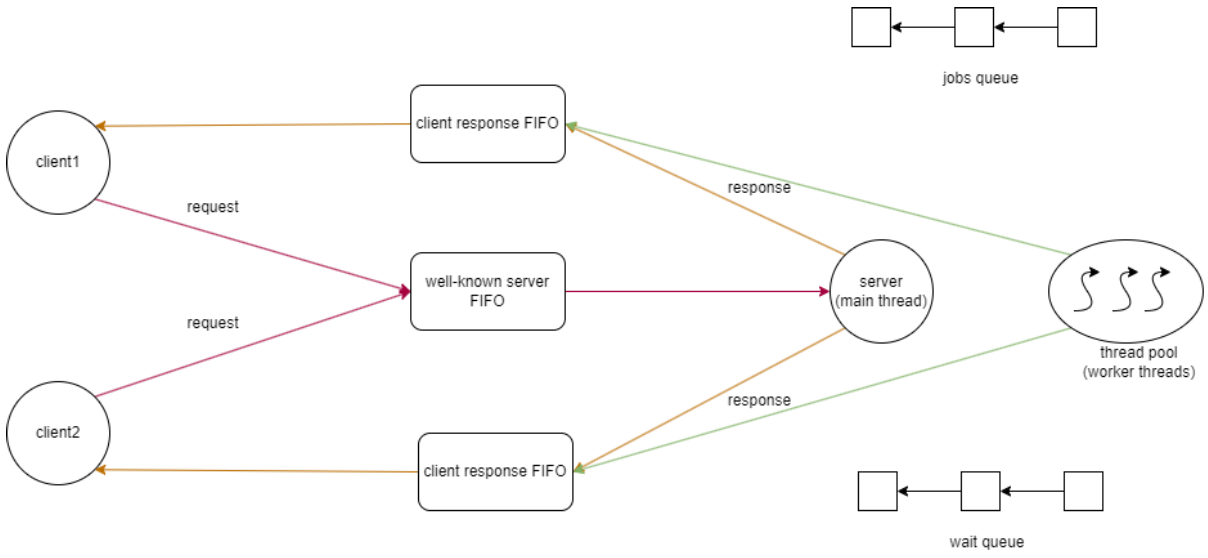
GTU Department of Computer Engineering
CSE 344 - Spring 2023
Hw4 Report

Emirkan Burak Yılmaz
1901042659

MybiboBox
A Concurrent File Access System

Contents

1 System Architecture



1.1 Connecting to the MyBiboBox Server

The communication between client and server is handled via unnamed pipes (FIFOs). There is one well known server FIFO which clients can write their request. For the connection to the server client prepares a packet which contains its PID, current working directory and the wait bit. The wait bit indicates whether the client prefers waiting in the wait queue or not. Client makes a connection request with the prepared packet to the main server via server FIFO and the main server identifies the client request, check the number of currently serving clients. According to that, main server prepares a response to indicate if it is possible to serve the client immediately or wait it a little bit. According to coming response from the main server, the client waits or gives up according to it connect option (connect/tryConnect) which is passed as command line argument to the client program. If the client is opened with connect option, it opens the client request FIFO and blocked until the server become available for taking further requests. If the client chose to give up, then it directly terminates without connecting to the server. For disconnection from the server, client can make quit request either intentionally or interrupted with SIGINT or SIGTERM signals, or the server is shut down and client automatically disconnect from the server.

1.2 Requests and Responses

All the necessary informations was send during the connection requests. Those information's are kept in waiting client queue for the waiting clients and in the currently serving client list for the serving clients. After the connection is established, clients make their request by writing the request header which contains the length of the data that will be send next and its PID for the identification. With that way all the clients write their request on the server FIFO and takes responses on their special response FIFO.

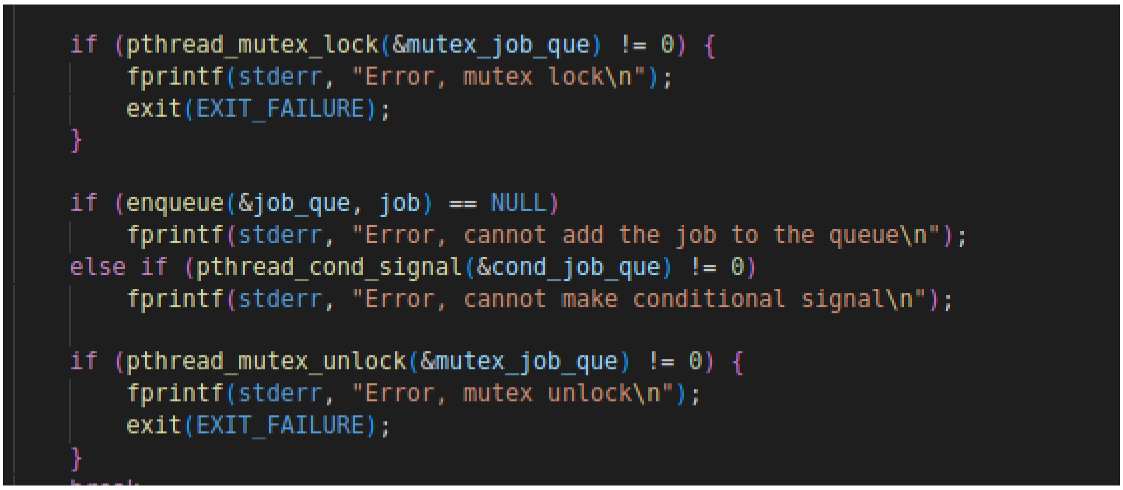
1.3 Main Thread and Worker Threads

Main threads are responsible for reading the client requests and take necessary actions. For the special requests such as connect, disconnect and kill are handled by the main thread and the rest are put in the jobs queue. The worker threads that are inside of the thread pool takes the jobs one by one and response the clients.

2 Synchronization

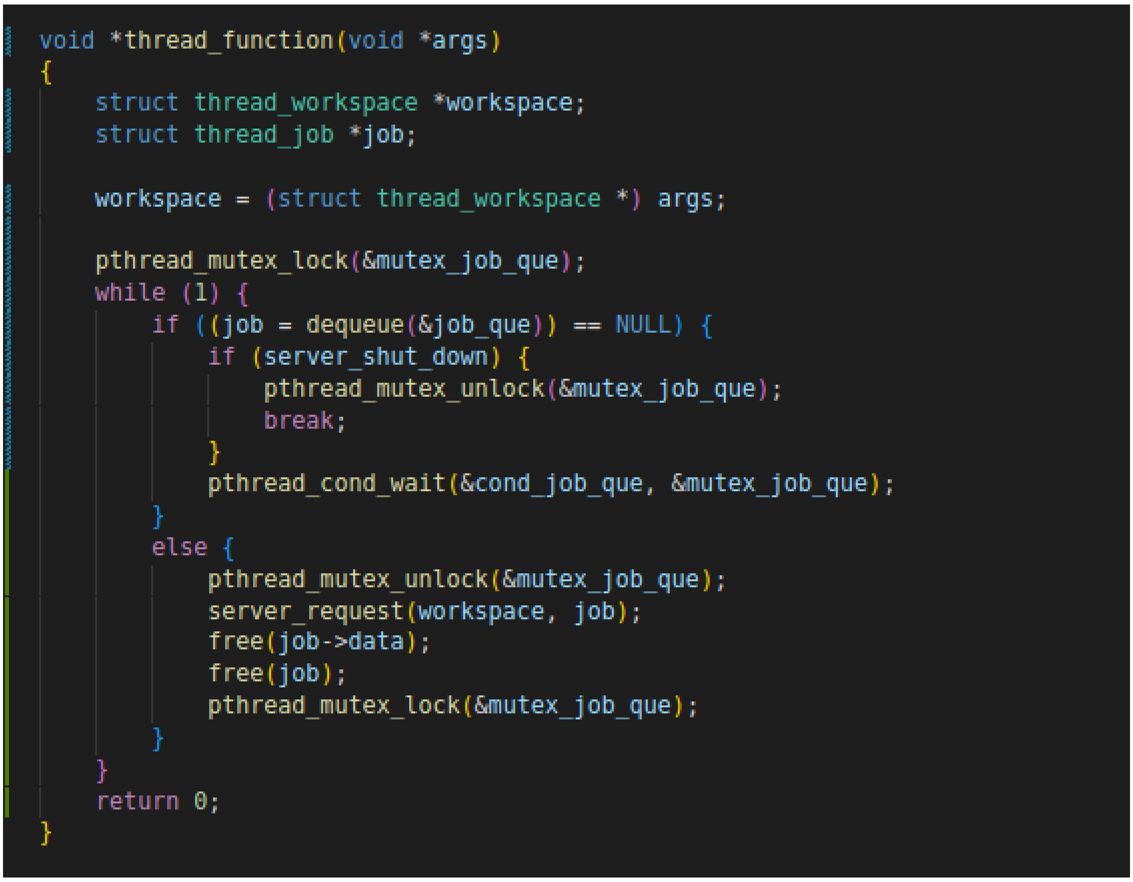
2.1 Job Queue

The race conditions for the job queue is prevented by using a mutex and a conditional variable.



```
    if (pthread_mutex_lock(&mutex_job_que) != 0) {
        fprintf(stderr, "Error, mutex lock\n");
        exit(EXIT_FAILURE);
    }

    if (enqueue(&job_que, job) == NULL)
        fprintf(stderr, "Error, cannot add the job to the queue\n");
    else if (pthread_cond_signal(&cond_job_que) != 0)
        fprintf(stderr, "Error, cannot make conditional signal\n");

    if (pthread_mutex_unlock(&mutex_job_que) != 0) {
        fprintf(stderr, "Error, mutex unlock\n");
        exit(EXIT_FAILURE);
    }
```

Main thread adds the jobs to the queue.

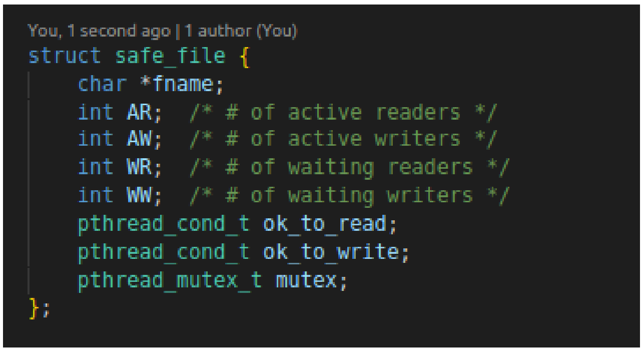


```
void *thread_function(void *args)
{
    struct thread_workspace *workspace;
    struct thread_job *job;

    workspace = (struct thread_workspace *) args;

    pthread_mutex_lock(&mutex_job_que);
    while (1) {
        if ((job = dequeue(&job_que)) == NULL) {
            if (server_shut_down) {
                pthread_mutex_unlock(&mutex_job_que);
                break;
            }
            pthread_cond_wait(&cond_job_que, &mutex_job_que);
        }
        else {
            pthread_mutex_unlock(&mutex_job_que);
            server_request(workspace, job);
            free(job->data);
            free(job);
            pthread_mutex_lock(&mutex_job_que);
        }
    }
    return 0;
}
```

Worker threads take the jobs from the queue.

2.2 Files in Server Directory

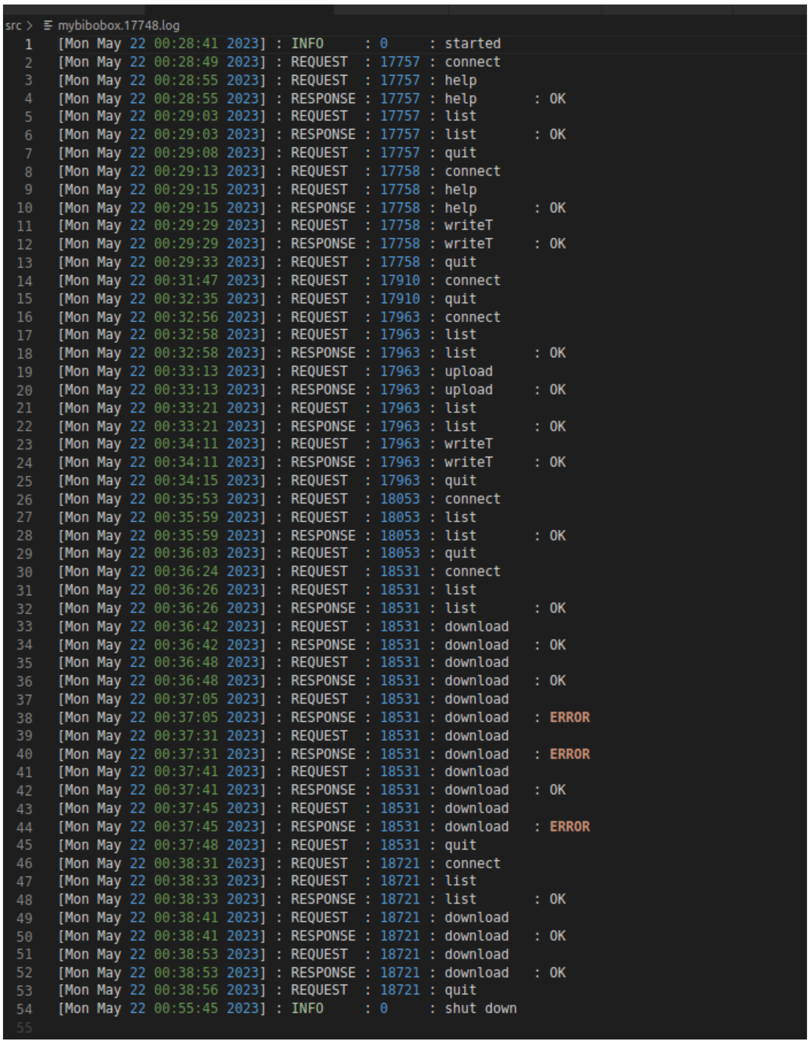
For enforcing mutual exclusion to prevent race conditions in the server directory, the readers-writers problem is followed. To integrate readers writers' approach, safe_file structure which contains necessary condition variables and mutexes is created.



```c
struct safe_file {
    char *fname;
    int AR;  /* # of active readers */
    int AW;  /* # of active writers */
    int WR;  /* # of waiting readers */
    int WW;  /* # of waiting writers */
    pthread_cond_t ok_to_read;
    pthread_cond_t ok_to_write;
    pthread_mutex_t mutex;
};
```

2.3 Log File

To provide mutual exclusion for the log file, single mutex is used.

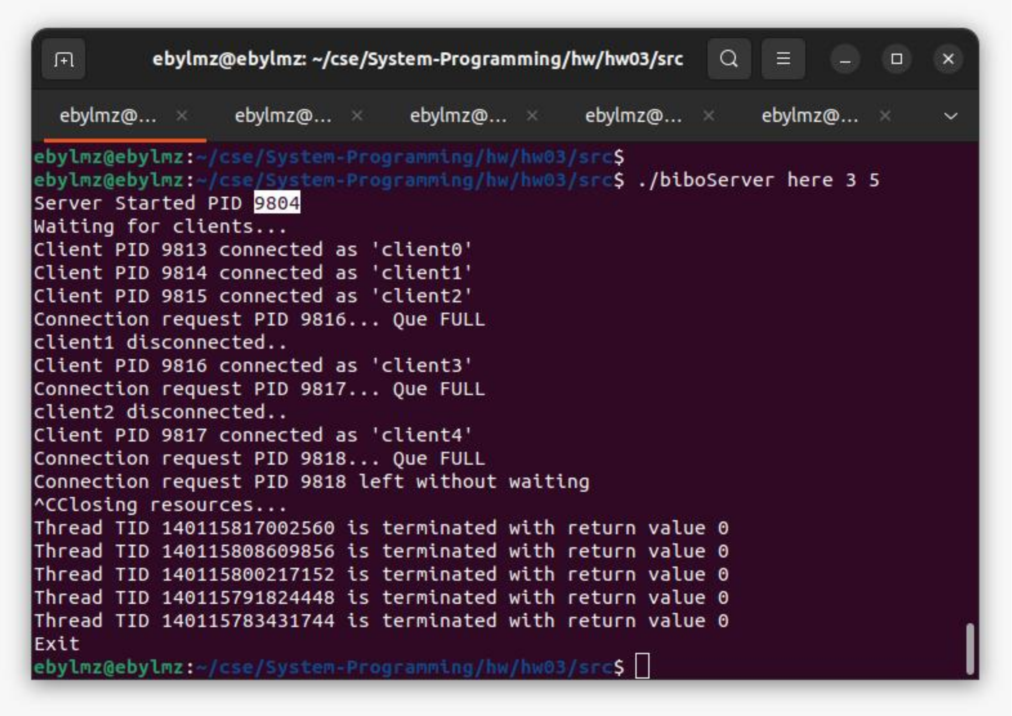


```
[Mon May 22 00:28:41 2023] : INFO     : 0     : started
[Mon May 22 00:28:49 2023] : REQUEST  : 17757 : connect
[Mon May 22 00:28:55 2023] : REQUEST  : 17757 : help
[Mon May 22 00:28:55 2023] : RESPONSE : 17757 : help       : OK
[Mon May 22 00:29:03 2023] : REQUEST  : 17757 : list
[Mon May 22 00:29:03 2023] : RESPONSE : 17757 : list       : OK
[Mon May 22 00:29:08 2023] : REQUEST  : 17757 : quit
[Mon May 22 00:29:13 2023] : REQUEST  : 17758 : connect
[Mon May 22 00:29:15 2023] : REQUEST  : 17758 : help
[Mon May 22 00:29:15 2023] : RESPONSE : 17758 : help       : OK
[Mon May 22 00:29:29 2023] : REQUEST  : 17758 : writeT
[Mon May 22 00:29:29 2023] : RESPONSE : 17758 : writeT     : OK
[Mon May 22 00:29:33 2023] : REQUEST  : 17758 : quit
[Mon May 22 00:31:47 2023] : REQUEST  : 17910 : connect
[Mon May 22 00:32:35 2023] : REQUEST  : 17910 : quit
[Mon May 22 00:32:56 2023] : REQUEST  : 17963 : connect
[Mon May 22 00:32:58 2023] : REQUEST  : 17963 : list
[Mon May 22 00:32:58 2023] : RESPONSE : 17963 : list       : OK
[Mon May 22 00:33:13 2023] : REQUEST  : 17963 : upload
[Mon May 22 00:33:13 2023] : RESPONSE : 17963 : upload     : OK
[Mon May 22 00:33:21 2023] : REQUEST  : 17963 : list
[Mon May 22 00:33:21 2023] : RESPONSE : 17963 : list       : OK
[Mon May 22 00:34:11 2023] : REQUEST  : 17963 : writeT
[Mon May 22 00:34:11 2023] : RESPONSE : 17963 : writeT     : OK
[Mon May 22 00:34:15 2023] : REQUEST  : 17963 : quit
[Mon May 22 00:35:53 2023] : REQUEST  : 18053 : connect
[Mon May 22 00:35:59 2023] : REQUEST  : 18053 : list
[Mon May 22 00:35:59 2023] : RESPONSE : 18053 : list       : OK
[Mon May 22 00:36:03 2023] : REQUEST  : 18053 : quit
[Mon May 22 00:36:24 2023] : REQUEST  : 18531 : connect
[Mon May 22 00:36:26 2023] : REQUEST  : 18531 : list
[Mon May 22 00:36:26 2023] : RESPONSE : 18531 : list       : OK
[Mon May 22 00:36:42 2023] : REQUEST  : 18531 : download
[Mon May 22 00:36:42 2023] : RESPONSE : 18531 : download   : OK
[Mon May 22 00:36:48 2023] : REQUEST  : 18531 : download
[Mon May 22 00:36:48 2023] : RESPONSE : 18531 : download   : OK
[Mon May 22 00:37:05 2023] : REQUEST  : 18531 : download
[Mon May 22 00:37:05 2023] : RESPONSE : 18531 : download   : ERROR
[Mon May 22 00:37:31 2023] : REQUEST  : 18531 : download
[Mon May 22 00:37:31 2023] : RESPONSE : 18531 : download   : ERROR
[Mon May 22 00:37:41 2023] : REQUEST  : 18531 : download
[Mon May 22 00:37:41 2023] : RESPONSE : 18531 : download   : OK
[Mon May 22 00:37:45 2023] : REQUEST  : 18531 : download
[Mon May 22 00:37:45 2023] : RESPONSE : 18531 : download   : ERROR
[Mon May 22 00:37:48 2023] : REQUEST  : 18531 : quit
[Mon May 22 00:38:31 2023] : REQUEST  : 18721 : connect
[Mon May 22 00:38:33 2023] : REQUEST  : 18721 : list
[Mon May 22 00:38:33 2023] : RESPONSE : 18721 : list       : OK
[Mon May 22 00:38:41 2023] : REQUEST  : 18721 : download
[Mon May 22 00:38:41 2023] : RESPONSE : 18721 : download   : OK
[Mon May 22 00:38:53 2023] : REQUEST  : 18721 : download
[Mon May 22 00:38:53 2023] : RESPONSE : 18721 : download   : OK
[Mon May 22 00:38:56 2023] : REQUEST  : 18721 : quit
[Mon May 22 00:55:45 2023] : INFO     : 0     : shut down
```

Sample log file

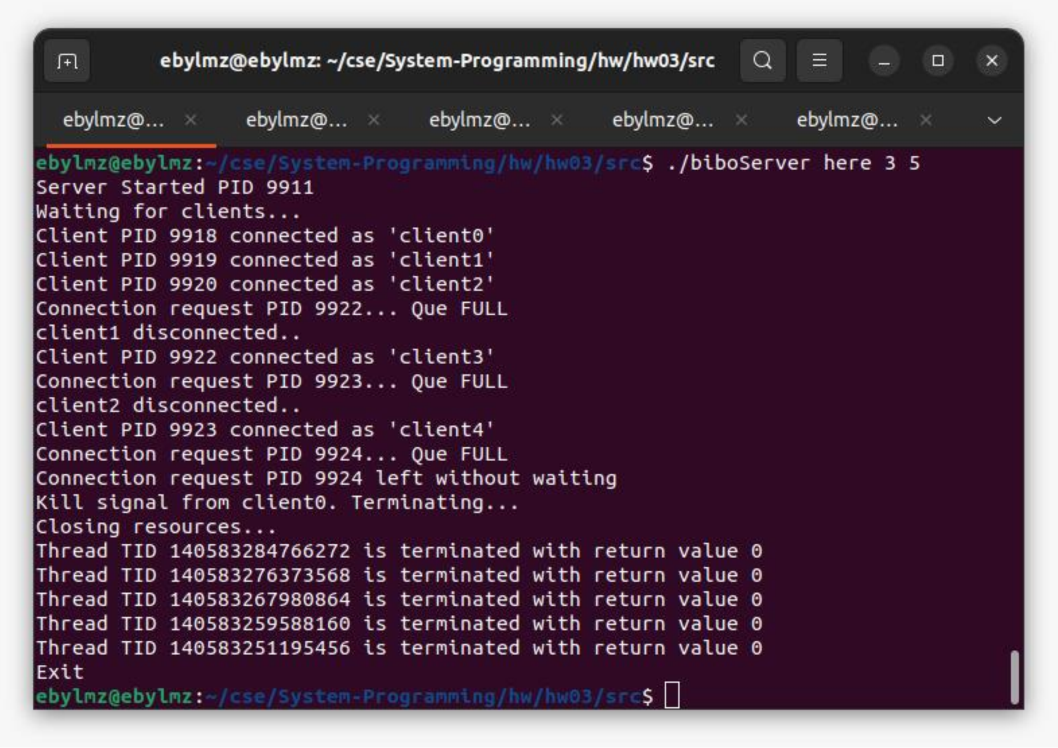
3 Test Cases and Results

3.1 Client Connection Options (connect/tryConnect)



```
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ./biboServer here 3 5
Server Started PID 9804
Waiting for clients...
Client PID 9813 connected as 'client0'
Client PID 9814 connected as 'client1'
Client PID 9815 connected as 'client2'
Connection request PID 9816... Que FULL
client1 disconnected..
Client PID 9816 connected as 'client3'
Connection request PID 9817... Que FULL
client2 disconnected..
Client PID 9817 connected as 'client4'
Connection request PID 9818... Que FULL
Connection request PID 9818 left without waiting
^CClosing resources...
Thread TID 140115817002560 is terminated with return value 0
Thread TID 140115808609856 is terminated with return value 0
Thread TID 140115800217152 is terminated with return value 0
Thread TID 140115791824448 is terminated with return value 0
Thread TID 140115783431744 is terminated with return value 0
Exit
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$
```

Termination with CTRL-C on server side



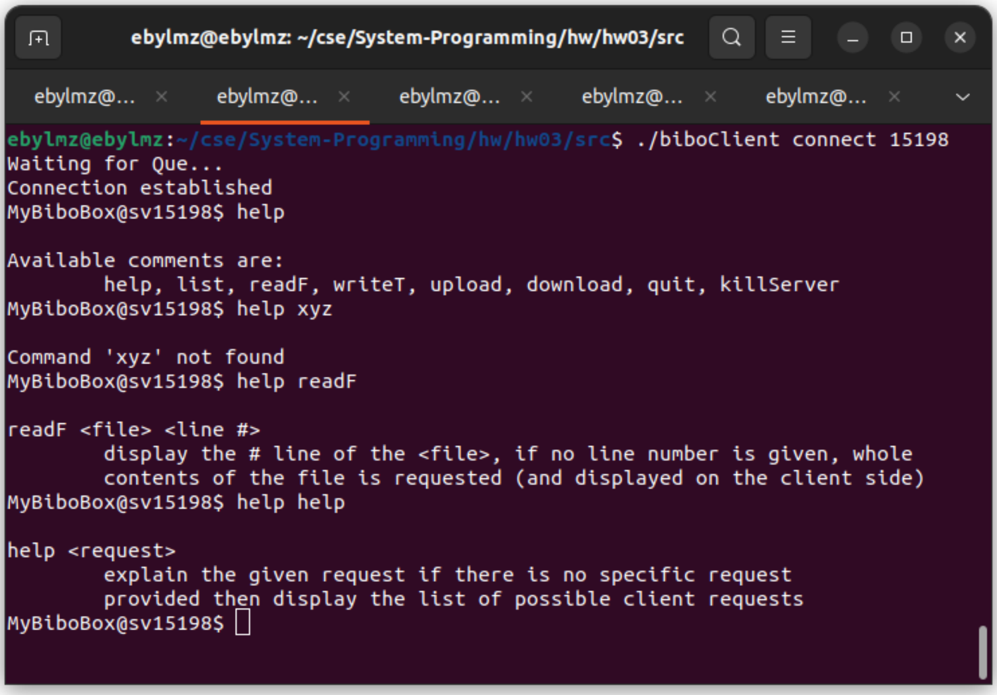
```
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ./biboServer here 3 5
Server Started PID 9911
Waiting for clients...
Client PID 9918 connected as 'client0'
Client PID 9919 connected as 'client1'
Client PID 9920 connected as 'client2'
Connection request PID 9922... Que FULL
client1 disconnected..
Client PID 9922 connected as 'client3'
Connection request PID 9923... Que FULL
client2 disconnected..
Client PID 9923 connected as 'client4'
Connection request PID 9924... Que FULL
Connection request PID 9924 left without waiting
Kill signal from client0. Terminating...
Closing resources...
Thread TID 140583284766272 is terminated with return value 0
Thread TID 140583276373568 is terminated with return value 0
Thread TID 140583267980864 is terminated with return value 0
Thread TID 140583259588160 is terminated with return value 0
Thread TID 140583251195456 is terminated with return value 0
Exit
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$
```

Termination with killServer request from the client

3.2 Basic User Commands

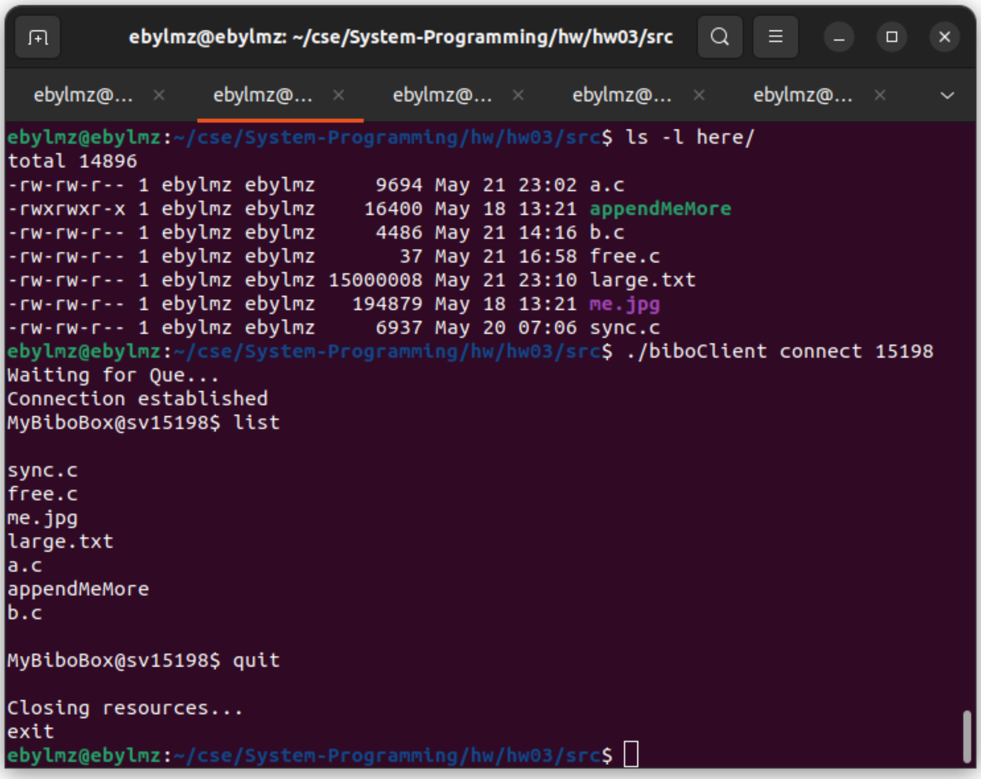
The user commands help, list, readF, writeT, upload, download, quit and killServer should satisfy the user needed.

3.2.1.1 help



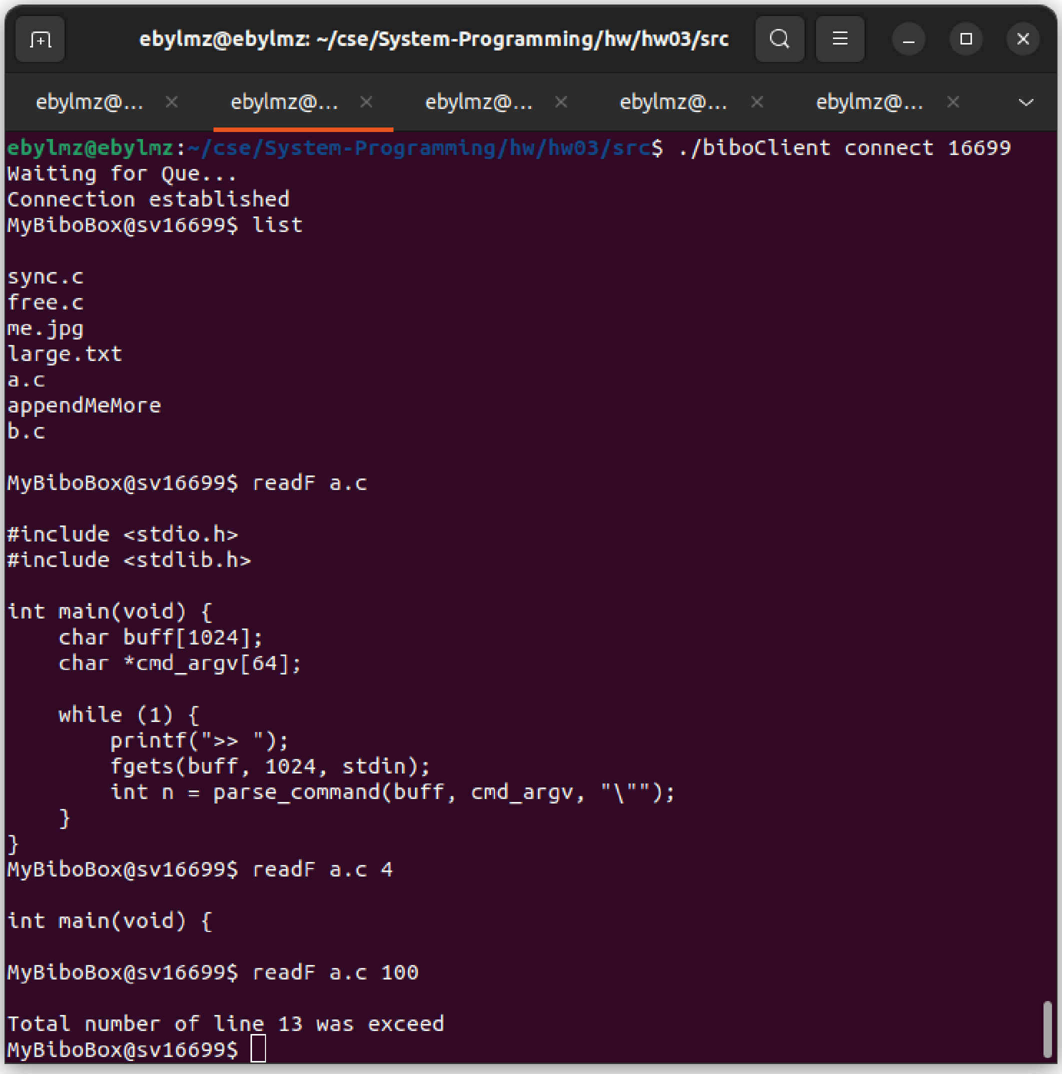
```
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ./biboClient connect 15198
Waiting for Que...
Connection established
MyBiboBox@sv15198$ help

Available comments are:
        help, list, readF, writeT, upload, download, quit, killServer
MyBiboBox@sv15198$ help xyz

Command 'xyz' not found
MyBiboBox@sv15198$ help readF

readF <file> <line #>
        display the # line of the <file>, if no line number is given, whole
        contents of the file is requested (and displayed on the client side)
MyBiboBox@sv15198$ help help

help <request>
        explain the given request if there is no specific request
        provided then display the list of possible client requests
MyBiboBox@sv15198$ 
```

3.2.1.2 list



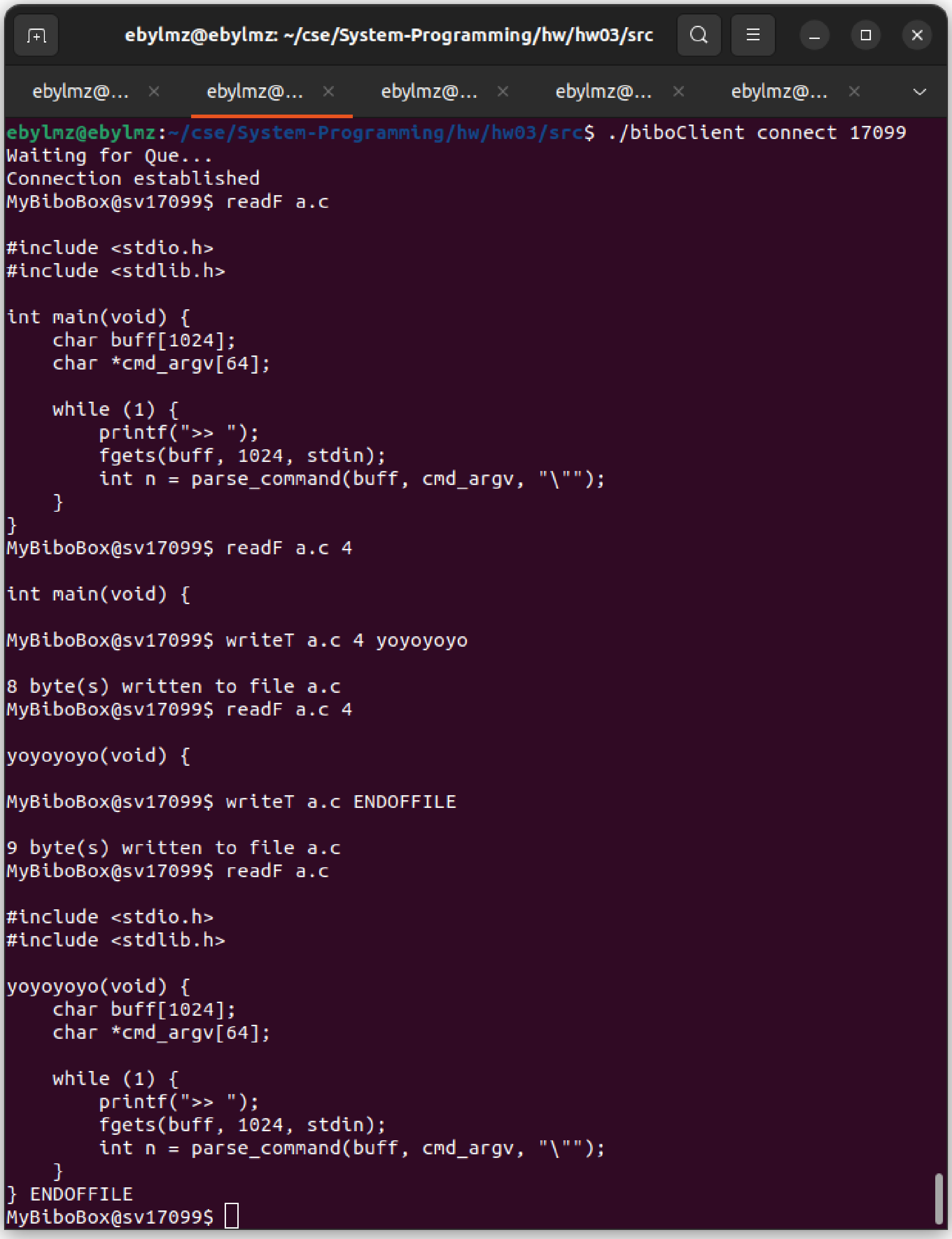
```
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ls -l here/
total 14896
-rw-rw-r-- 1 ebylmz ebylmz     9694 May 21 23:02 a.c
-rwxrwxr-x 1 ebylmz ebylmz    16400 May 18 13:21 appendMeMore
-rw-rw-r-- 1 ebylmz ebylmz     4486 May 21 14:16 b.c
-rw-rw-r-- 1 ebylmz ebylmz       37 May 21 16:58 free.c
-rw-rw-r-- 1 ebylmz ebylmz 15000008 May 21 23:10 large.txt
-rw-rw-r-- 1 ebylmz ebylmz   194879 May 18 13:21 me.jpg
-rw-rw-r-- 1 ebylmz ebylmz     6937 May 20 07:06 sync.c
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ./biboClient connect 15198
Waiting for Que...
Connection established
MyBiboBox@sv15198$ list

sync.c
free.c
me.jpg
large.txt
a.c
appendMeMore
b.c

MyBiboBox@sv15198$ quit

Closing resources...
exit
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ 
```

3.2.1.3 readF



```
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ./biboClient connect 16699
Waiting for Que...
Connection established
MyBiboBox@sv16699$ list

sync.c
free.c
me.jpg
large.txt
a.c
appendMeMore
b.c

MyBiboBox@sv16699$ readF a.c

#include <stdio.h>
#include <stdlib.h>

int main(void) {
    char buff[1024];
    char *cmd_argv[64];

    while (1) {
        printf(">> ");
        fgets(buff, 1024, stdin);
        int n = parse_command(buff, cmd_argv, "\"");
    }
}
MyBiboBox@sv16699$ readF a.c 4

int main(void) {

MyBiboBox@sv16699$ readF a.c 100

Total number of line 13 was exceed
MyBiboBox@sv16699$ 
```

3.2.1.4 writeT



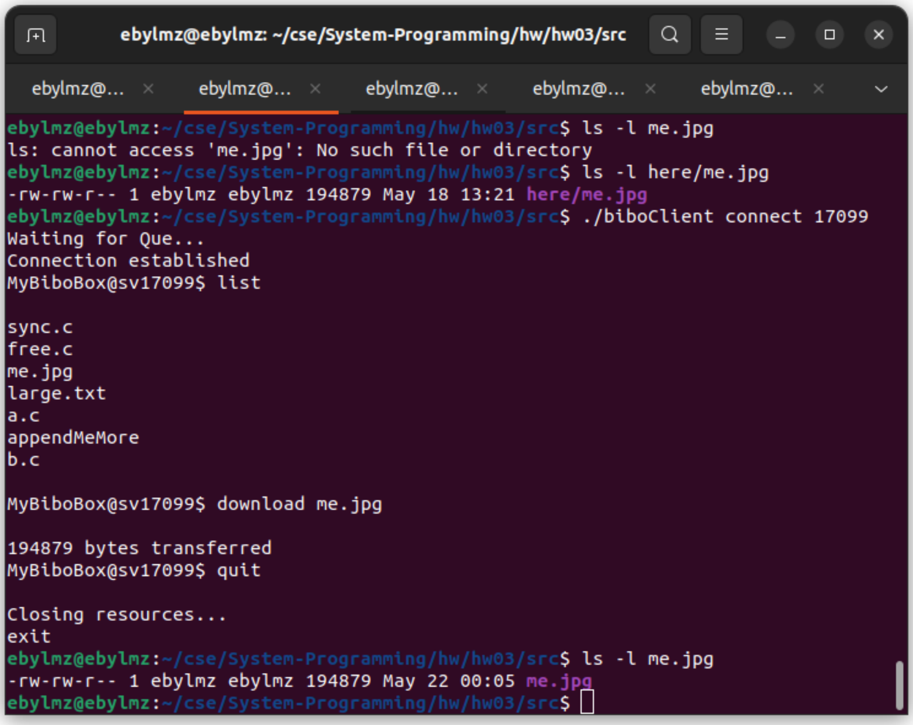
ebylmz@ebylmz: ~/cse/System-Programming/hw/hw03/src
ebylmz@... ebylmz@... ebylmz@... ebylmz@... ebylmz@...

```
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ./biboClient connect 17099
Waiting for Que...
Connection established
MyBiboBox@sv17099$ readF a.c

#include <stdio.h>
#include <stdlib.h>

int main(void) {
    char buff[1024];
    char *cmd_argv[64];

    while (1) {
        printf(">> ");
        fgets(buff, 1024, stdin);
        int n = parse_command(buff, cmd_argv, "\"");
    }
}
MyBiboBox@sv17099$ readF a.c 4

int main(void) {

MyBiboBox@sv17099$ writeT a.c 4 yoyoyoyo

8 byte(s) written to file a.c
MyBiboBox@sv17099$ readF a.c 4

yoyoyoyo(void) {

MyBiboBox@sv17099$ writeT a.c ENDOFFILE

9 byte(s) written to file a.c
MyBiboBox@sv17099$ readF a.c

#include <stdio.h>
#include <stdlib.h>

yoyoyoyo(void) {
    char buff[1024];
    char *cmd_argv[64];

    while (1) {
        printf(">> ");
        fgets(buff, 1024, stdin);
        int n = parse_command(buff, cmd_argv, "\"");
    }
} ENDOFFILE
MyBiboBox@sv17099$ 
```

Unfortunately, I writeT hasn’t detect the string in the quotes yet :(

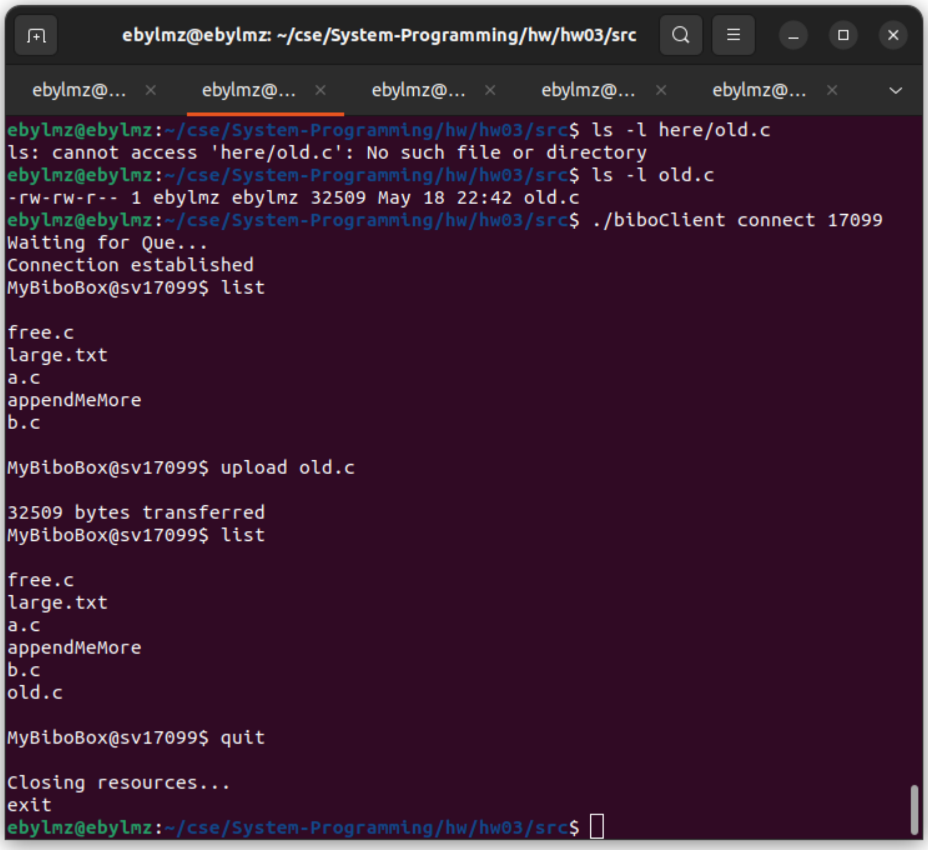
If the given line is exceeding the current number of lines, then it appends empty lines till the target line is reached and then write the given string.

3.2.1.5 download



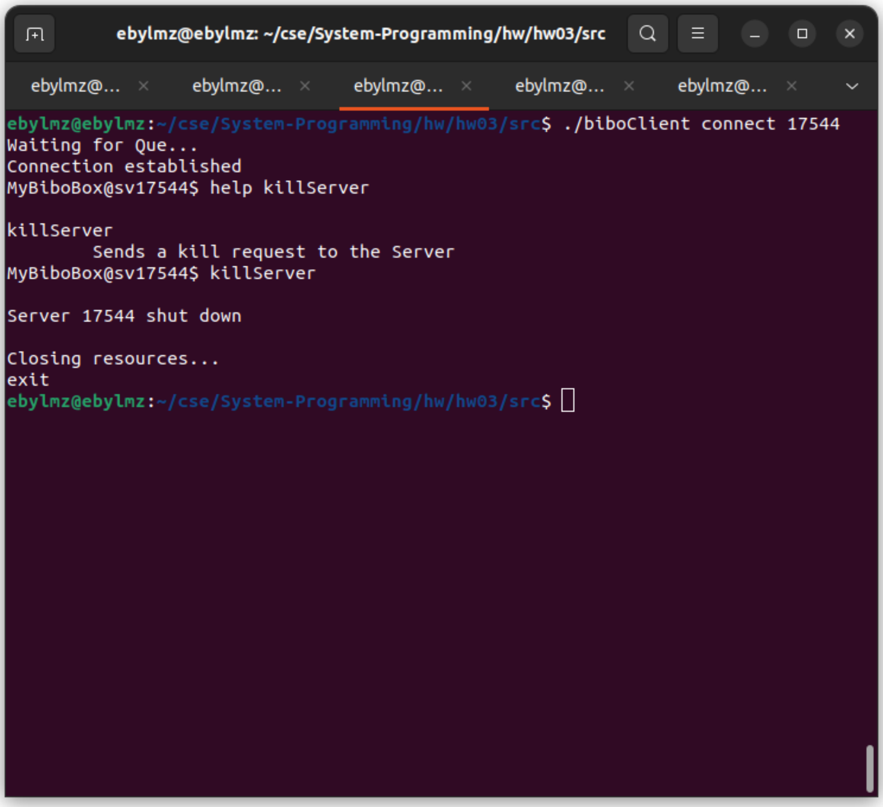
```
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ls -l me.jpg
ls: cannot access 'me.jpg': No such file or directory
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ls -l here/me.jpg
-rw-rw-r-- 1 ebylmz ebylmz 194879 May 18 13:21 here/me.jpg
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ./biboClient connect 17099
Waiting for Que...
Connection established
MyBiboBox@sv17099$ list

sync.c
free.c
me.jpg
large.txt
a.c
appendMeMore
b.c

MyBiboBox@sv17099$ download me.jpg

194879 bytes transferred
MyBiboBox@sv17099$ quit

Closing resources...
exit
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ls -l me.jpg
-rw-rw-r-- 1 ebylmz ebylmz 194879 May 22 00:05 me.jpg
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ 
```

3.2.1.6 upload

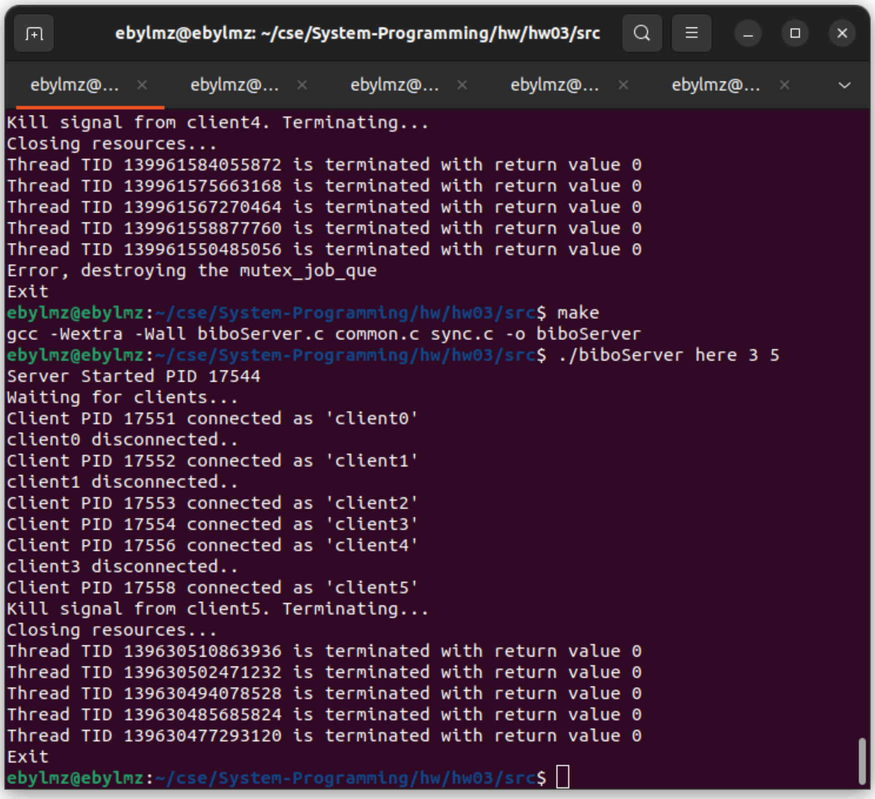


```
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ls -l here/old.c
ls: cannot access 'here/old.c': No such file or directory
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ls -l old.c
-rw-rw-r-- 1 ebylmz ebylmz 32509 May 18 22:42 old.c
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ./biboClient connect 17099
Waiting for Que...
Connection established
MyBiboBox@sv17099$ list

free.c
large.txt
a.c
appendMeMore
b.c

MyBiboBox@sv17099$ upload old.c

32509 bytes transferred
MyBiboBox@sv17099$ list

free.c
large.txt
a.c
appendMeMore
b.c
old.c

MyBiboBox@sv17099$ quit

Closing resources...
exit
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ 
```

3.2.1.7 killServer



```
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ./biboClient connect 17544
Waiting for Que...
Connection established
MyBiboBox@sv17544$ help killServer

killServer
        Sends a kill request to the Server
MyBiboBox@sv17544$ killServer

Server 17544 shut down

Closing resources...
exit
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ 
```

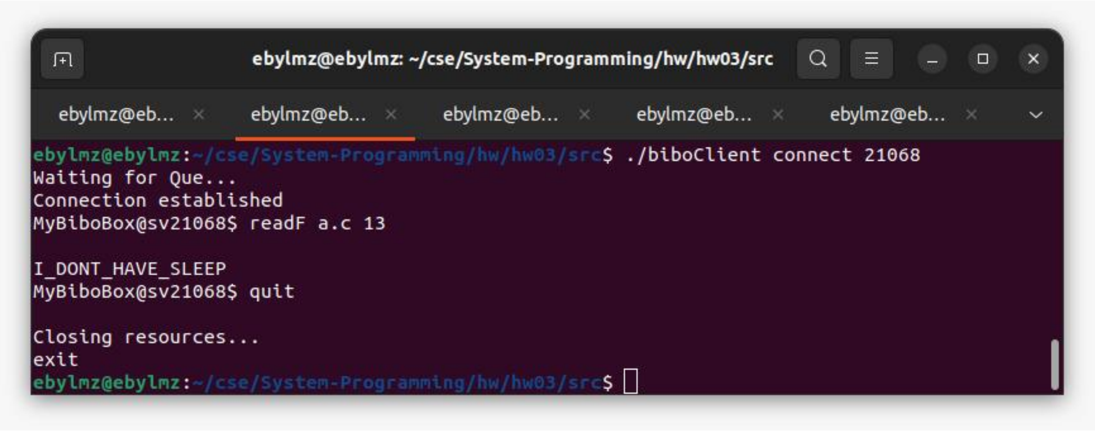


```
Kill signal from client4. Terminating...
Closing resources...
Thread TID 139961584055872 is terminated with return value 0
Thread TID 139961575663168 is terminated with return value 0
Thread TID 139961567270464 is terminated with return value 0
Thread TID 139961558877760 is terminated with return value 0
Thread TID 139961550485056 is terminated with return value 0
Error, destroying the mutex_job_que
Exit
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ make
gcc -Wextra -Wall biboServer.c common.c sync.c -o biboServer
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ./biboServer here 3 5
Server Started PID 17544
Waiting for clients...
Client PID 17551 connected as 'client0'
client0 disconnected..
Client PID 17552 connected as 'client1'
client1 disconnected..
Client PID 17553 connected as 'client2'
Client PID 17554 connected as 'client3'
Client PID 17556 connected as 'client4'
client3 disconnected..
Client PID 17558 connected as 'client5'
Kill signal from client5. Terminating...
Closing resources...
Thread TID 139630510863936 is terminated with return value 0
Thread TID 139630502471232 is terminated with return value 0
Thread TID 139630494078528 is terminated with return value 0
Thread TID 139630485685824 is terminated with return value 0
Thread TID 139630477293120 is terminated with return value 0
Exit
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ 
```

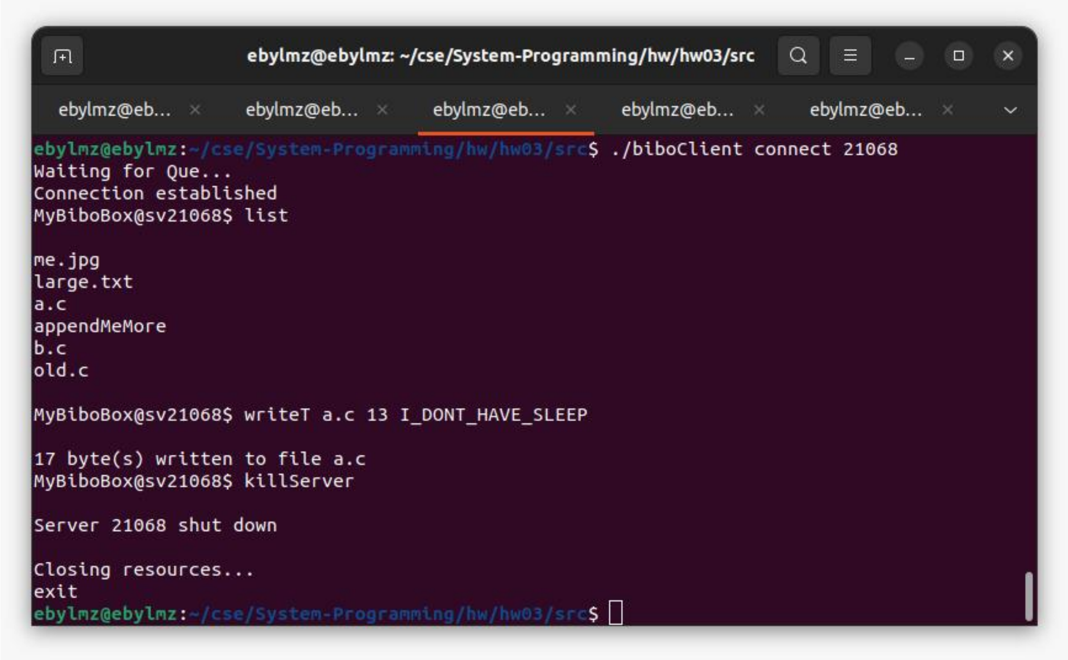
3.3 Providing Mutual Exclusion and Preventing Race Conditions

- More than one process may want to write log.
- readF and writeT requests may happen on the same file at the same time.
- During readF or writeT requests download or upload request may occur.
- During upload request list request may happen.

Multiple reader can access the same resource, but only single writer can access the resource.

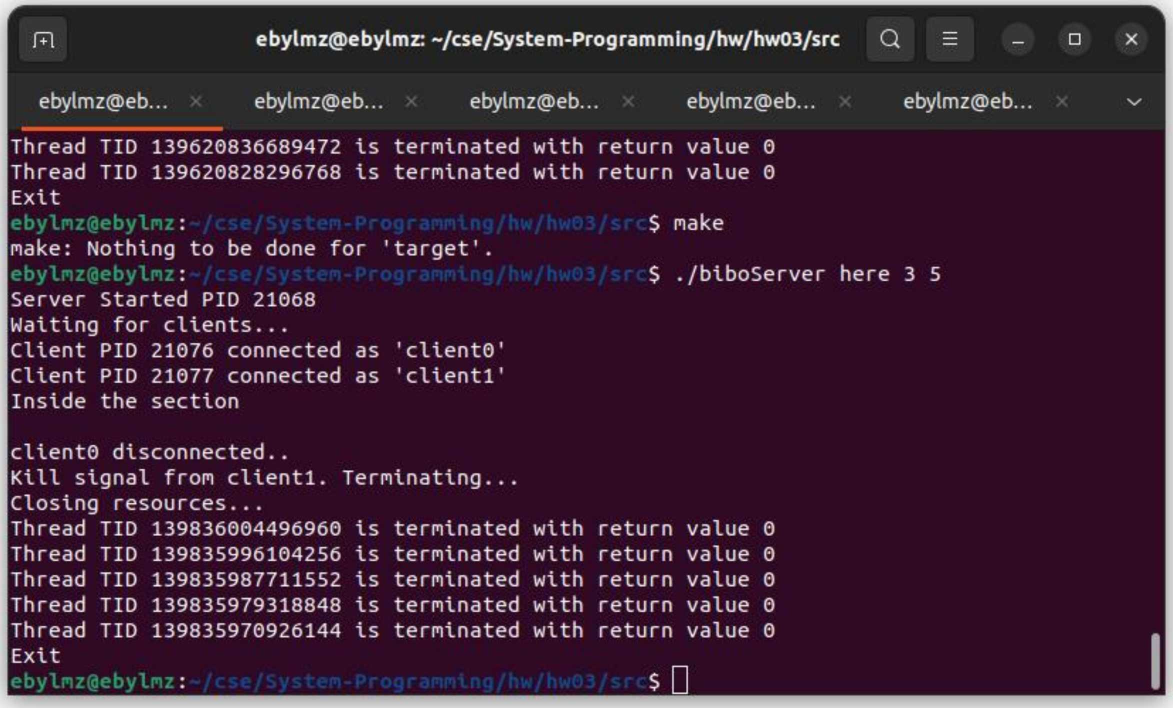


```
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ./biboClient connect 21068
Waiting for Que...
Connection established
MyBiboBox@sv21068$ readF a.c 13

I_DONT_HAVE_SLEEP
MyBiboBox@sv21068$ quit

Closing resources...
exit
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ 
```

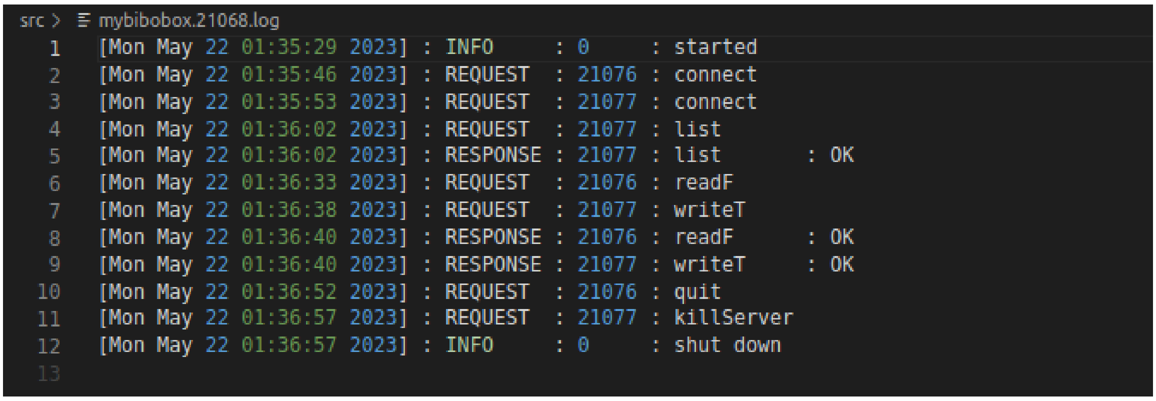


```
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ./biboClient connect 21068
Waiting for Que...
Connection established
MyBiboBox@sv21068$ list

me.jpg
large.txt
a.c
appendMeMore
b.c
old.c

MyBiboBox@sv21068$ writeT a.c 13 I_DONT_HAVE_SLEEP

17 byte(s) written to file a.c
MyBiboBox@sv21068$ killServer

Server 21068 shut down

Closing resources...
exit
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ 
```

A reader and writer trying to access the same file.



```
Thread TID 139620836689472 is terminated with return value 0
Thread TID 139620828296768 is terminated with return value 0
Exit
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ make
make: Nothing to be done for 'target'.
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ./biboServer here 3 5
Server Started PID 21068
Waiting for clients...
Client PID 21076 connected as 'client0'
Client PID 21077 connected as 'client1'
Inside the section

client0 disconnected..
Kill signal from client1. Terminating...
Closing resources...
Thread TID 139836004496960 is terminated with return value 0
Thread TID 139835996104256 is terminated with return value 0
Thread TID 139835987711552 is terminated with return value 0
Thread TID 139835979318848 is terminated with return value 0
Thread TID 139835970926144 is terminated with return value 0
Exit
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ 
```

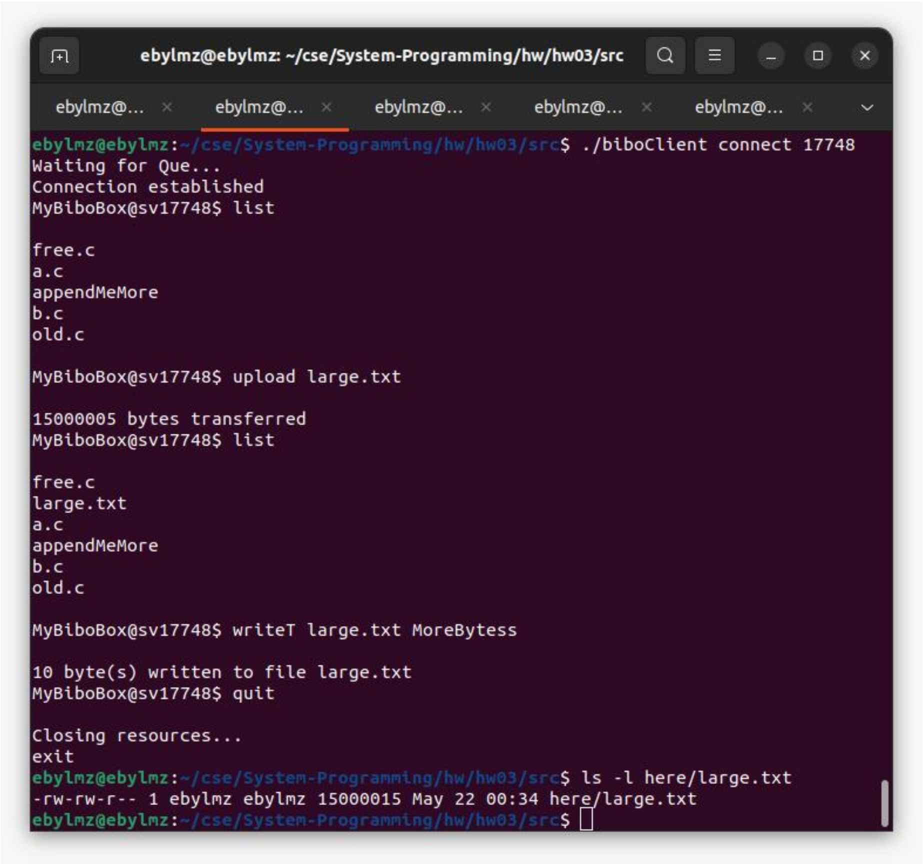


```
[Mon May 22 01:35:29 2023] : INFO     : 0     : started
[Mon May 22 01:35:46 2023] : REQUEST  : 21076 : connect
[Mon May 22 01:35:53 2023] : REQUEST  : 21077 : connect
[Mon May 22 01:36:02 2023] : REQUEST  : 21077 : list
[Mon May 22 01:36:02 2023] : RESPONSE : 21077 : list        : OK
[Mon May 22 01:36:33 2023] : REQUEST  : 21076 : readF
[Mon May 22 01:36:38 2023] : REQUEST  : 21077 : writeT
[Mon May 22 01:36:40 2023] : RESPONSE : 21076 : readF       : OK
[Mon May 22 01:36:40 2023] : RESPONSE : 21077 : writeT      : OK
[Mon May 22 01:36:52 2023] : REQUEST  : 21076 : quit
[Mon May 22 01:36:57 2023] : REQUEST  : 21077 : killServer
[Mon May 22 01:36:57 2023] : INFO     : 0     : shut down
```

Readers get the lock and writer block until it finishes his read.

3.4 Large Files

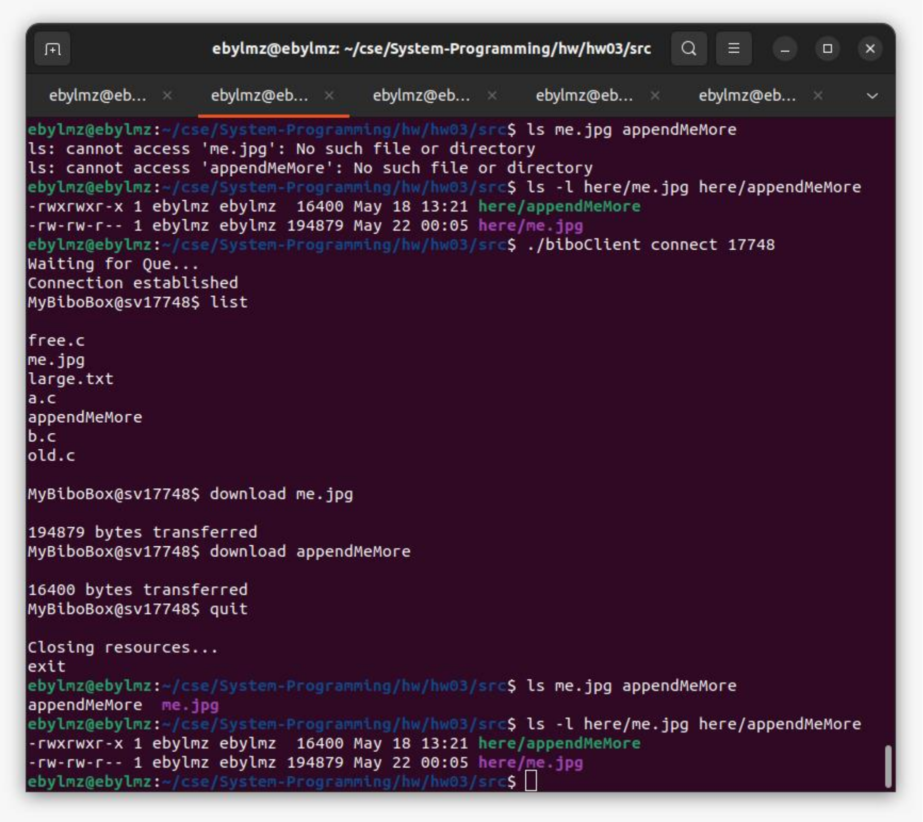
The user commands such as upload, download, readF and writeT should be able to handle large files (i.e., > 10 MB)



```
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ./biboClient connect 17748
Waiting for Que...
Connection established
MyBiboBox@sv17748$ list

free.c
a.c
appendMeMore
b.c
old.c

MyBiboBox@sv17748$ upload large.txt

15000005 bytes transferred
MyBiboBox@sv17748$ list

free.c
large.txt
a.c
appendMeMore
b.c
old.c

MyBiboBox@sv17748$ writeT large.txt MoreBytess

10 byte(s) written to file large.txt
MyBiboBox@sv17748$ quit

Closing resources...
exit
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ls -l here/large.txt
-rw-rw-r-- 1 ebylmz ebylmz 15000015 May 22 00:34 here/large.txt
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$
```

3.5 Different File Formats (text, binary)

Different file formats (i.e., text, binary) should be handled.

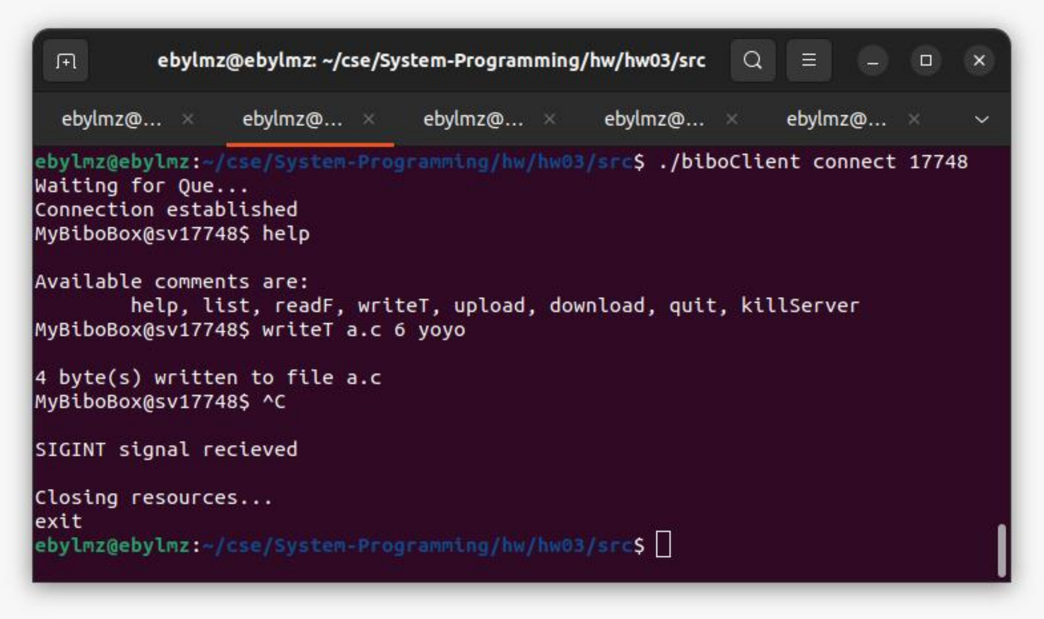


```
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ls me.jpg appendMeMore
ls: cannot access 'me.jpg': No such file or directory
ls: cannot access 'appendMeMore': No such file or directory
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ls -l here/me.jpg here/appendMeMore
-rwxrwxr-x 1 ebylmz ebylmz  16400 May 18 13:21 here/appendMeMore
-rw-rw-r-- 1 ebylmz ebylmz 194879 May 22 00:05 here/me.jpg
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ./biboClient connect 17748
Waiting for Que...
Connection established
MyBiboBox@sv17748$ list

free.c
me.jpg
large.txt
a.c
appendMeMore
b.c
old.c

MyBiboBox@sv17748$ download me.jpg

194879 bytes transferred
MyBiboBox@sv17748$ download appendMeMore

16400 bytes transferred
MyBiboBox@sv17748$ quit

Closing resources...
exit
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ls me.jpg appendMeMore
appendMeMore  me.jpg
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ls -l here/me.jpg here/appendMeMore
-rwxrwxr-x 1 ebylmz ebylmz  16400 May 18 13:21 here/appendMeMore
-rw-rw-r-- 1 ebylmz ebylmz 194879 May 22 00:05 here/me.jpg
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$
```

3.6 Signals

Both server and client should be able to handle the signals properly.

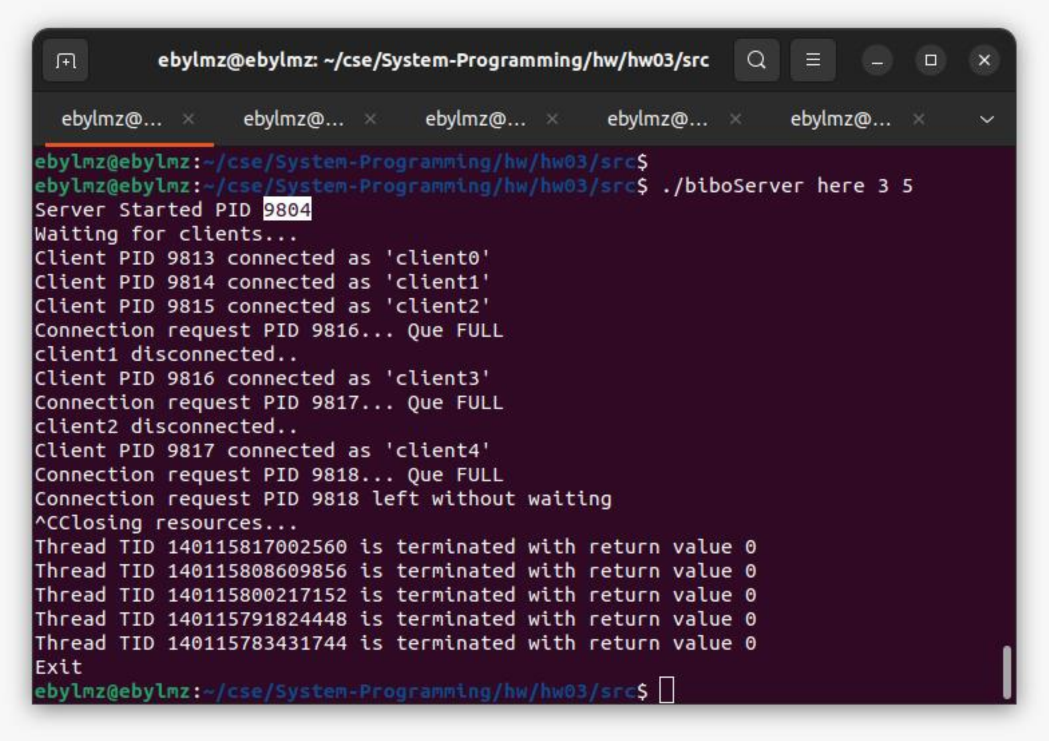
3.6.1 SIGINT/SIGTERM in Client



```
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ./biboClient connect 17748
Waiting for Que...
Connection established
MyBiboBox@sv17748$ help

Available comments are:
        help, list, readF, writeT, upload, download, quit, killServer
MyBiboBox@sv17748$ writeT a.c 6 yoyo

4 byte(s) written to file a.c
MyBiboBox@sv17748$ ^C

SIGINT signal recieved

Closing resources...
exit
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ 
```

Resources are closed properly and exited gracefully.

3.6.2 SIGINT/SIGTERM in Server



```
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ ./biboServer here 3 5
Server Started PID 9804
Waiting for clients...
Client PID 9813 connected as 'client0'
Client PID 9814 connected as 'client1'
Client PID 9815 connected as 'client2'
Connection request PID 9816... Que FULL
client1 disconnected..
Client PID 9816 connected as 'client3'
Connection request PID 9817... Que FULL
client2 disconnected..
Client PID 9817 connected as 'client4'
Connection request PID 9818... Que FULL
Connection request PID 9818 left without waiting
^CClosing resources...
Thread TID 140115817002560 is terminated with return value 0
Thread TID 140115808609856 is terminated with return value 0
Thread TID 140115800217152 is terminated with return value 0
Thread TID 140115791824448 is terminated with return value 0
Thread TID 140115783431744 is terminated with return value 0
Exit
ebylmz@ebylmz:~/cse/System-Programming/hw/hw03/src$ 
```

Threads are joined so that they can be detached, resources are closed properly and exited gracefully.