

SAMPLE INPUT/OUTPUT and UNIT TESTS

Submitter name: Keshav Krishna
Roll No.: 2019csb1224
Course: CSL303

1. What does this program do

The program tries to simulate the working of a dispatcher in an OS. Client sends requests to a multi threaded socket server, from which a dispatcher takes the requests, and checks whether it falls under the limits of thread number, memory or number of files open. If it is under the limits, DLL function is invoked. The client sends which function to invoke with which arguments and the server computes the result and returns the answer to the client. The client program ends there while the server continues to listen to more requests.

2. A description of how this program works (i.e. its logic)

First the user enters the command to start the server. This starts the server socket function. This in turn calls the make named socket function, which is used to create, in this case, a server socket. The returned server socket is then used to accept client requests. An infinite loop is created, which keeps listening for client requests. If a client request comes, a new worker thread is created. Worker thread creates a pthread, which then calls the dispatcher logic. In dispatcher logic, the server reads input sent by the client. It extracts the library, the function to be executed and its arguments. Arguments of functions are only integers, not fractional numbers. It then uses dlopen to open the library, failure in this case returns an error message. Then dlsym looks the value of the string in the library opened by dlopen. Finally the value of computation is written to the socket. After the computation is over, socket is closed, and pthread is exited.

There are also limits put on number of threads, memory and files. Thread limit is implemented using a boolean array that remains true at a particular index for entirety of thread execution and turns false, when the pthread ends. At every instance before creating the worker thread, false values are searched in the boolean array and if no false value exists, the thread request is denied and the server keeps waiting for more reuests.

To set file limit, setrlimit is used with RLIMIT_NOFILE and to set memory, setrlimit is used with RLIMIT_AS.

When file limit is reached, client request ends and server continues.

3. How to compile and run this program

First the file name test.c is compiled: `gcc server_client.c -lpthread -o ipc_demo -ldl`

Then it is executed: `./ipc_demo server ./cs303_sock 7 100000000 3`

here

`file_limit = 7`

`memory = 100000000`

`thread_limit = 3`

The format is executable file followed by sever/client followed by socket file.

If server is chosen, then following three inputs are file_limit, memory_limit, thread_limit.

Then client requests are executed:

`./ipc_demo client ./cs303_sock "/lib/x86_64-linux-gnu/libm.so.6 cos 2"`

format of the string in client request is <library path function argument>

constraints:

thread count ≥ 1 && < 100
file count ≥ 7
memory ≥ 1000000000

DLL:

library implemented: `/lib/x86_64-linux-gnu/libm.so.6`

Functions: cos, sin, tan, abs, ceil, floor, log (single argument functions of math.h)

4. Sample Outputs

server:

```
keshav@keshav-Lenovo-ideapad:~/Documents/cs303/assign1$ gcc test.c -lpthread -o ipc_demo -ldl
test.c: In function 'create_worker_thread':
test.c:222:29: warning: passing argument 3 of 'pthread_create' from incompatible pointer type [-Wincompatible-pointer-types]
222 |         dispatcher_logic,
    |         ^~~~~~
    |         |
    |         void (*)(void *)
In file included from test.c:9:
/usr/include/pthread.h:200:15: note: expected 'void * (*)(void *)' but argument is of type 'void (*)(void *)'
200 |     void *(*__start_routine) (void *),
    |     ~~~~~^~~~~~
keshav@keshav-Lenovo-ideapad:~/Documents/cs303/assign1$ ./ipc_demo server
./cs303_sock 7 1000000000 2
Creating AF_LOCAL socket at path ./cs303_sock
An old socket file exists, removing it.
Listening for client connections...

Waiting for incoming connections...
SERVER: Creating a worker thread.
Waiting for incoming connections...
SERVER: dispatcher_logic: starting
SERVER: Received from client: /lib/x86_64-linux-gnu/libm.so.6 cos 2
SERVER: dispatcher_logic: Done. Worker thread terminating.
```

client:

```
keshav@keshav-Lenovo-ideapad:~/Documents/cs303/assign1$ ./ipc_demo client
./cs303_sock "/lib/x86_64-linux-gnu/libm.so.6 cos 2"
Creating AF_LOCAL socket at path ./cs303_sock
CLIENT: Connect to server, about to write some stuff...
CLIENT: Received from server:: -0.416147
CLIENT: Processing done, about to exit...
```

server(with thread limit reached):

```
keshav@keshav-Lenovo-ideapad:~/Documents/cs303/assign1$ ./ipc_demo server
./cs303_sock 7 1000000000 0
Creating AF_LOCAL socket at path ./cs303_sock
An old socket file exists, removing it.
Listening for client connections...

Waiting for incoming connections...
Failed to create worker thread due to thread limit reached. Continuing to next.
Waiting for incoming connections...
```

5. Description of unit tests:

General format to run unit tests:

start server:

```
gcc unit_test.c -lpthread -o ipc_demo -ldl
./ipc_demo server <unit test no(default is 1)>
```

Unit test 1:

It checks normal compiling and execution

Unit test 2:

In this thread limit is set to 1

So when server is started :

```
keshav@keshav-Lenovo-ideapad:~/Documents/cs303/assign1$ ./ipc_demo server
Creating AF_LOCAL socket at path ./cs303_sock
An old socket file exists, removing it.
Listening for client connections...

Waiting for incoming connections...
```

Two client requests are sent

First one completes:

```
keshav@keshav-Lenovo-ideapad:~/Documents/cs303/assign1$ ./ipc_demo client
Creating AF_LOCAL socket at path ./cs303_sock
CLIENT: Connect to server, about to write some stuff...
CLIENT: Received from server:: -0.416147
CLIENT: Processing done, about to exit...
```

Second one does not, it is denied access and so it is left as it is:

```
keshav@keshav-Lenovo-ideapad:~/Documents/cs303/assign1$ ./ipc_demo client
Creating AF_LOCAL socket at path ./cs303_sock
CLIENT: Connect to server, about to write some stuff...
█
```

Server displays the messages:

```
keshav@keshav-Lenovo-ideapad:~/Documents/cs303/assign1$ ./ipc_demo server
Creating AF_LOCAL socket at path ./cs303_sock
An old socket file exists, removing it.
Listening for client connections...

Waiting for incoming connections...
SERVER: Creating a worker thread.
Waiting for incoming connections...
SERVER: dispatcher_logic: starting
Failed to create worker thread due to thread limit reached. Continuing to next.
Waiting for incoming connections...
SERVER: Received from client: /lib/x86_64-linux-gnu/libm.so.6 cos 2
SERVER: dispatcher_logic: Done. Worker thread terminating.
```

For second client request, it displays:

Failed to create worker thread due to thread limit reached. Continuing to next.

Waiting for incoming connections...

If a new request comes, then, it gets executed.

```

keshav@keshav-Lenovo-ideapad:~/Documents/cs303/assign1$ ./ipc_demo server
Creating AF_LOCAL socket at path ./cs303_sock
An old socket file exists, removing it.
Listening for client connections...

Waiting for incoming connections...
SERVER: Creating a worker thread.
Waiting for incoming connections...
SERVER: dispatcher_logic: starting
Failed to create worker thread due to thread limit reached. Continuing to
next.
Waiting for incoming connections...
SERVER: Received from client: /lib/x86_64-linux-gnu/libm.so.6 cos 2
SERVER: dispatcher_logic: Done. Worker thread terminating.
SERVER: Creating a worker thread.
Waiting for incoming connections...
SERVER: dispatcher_logic: starting
SERVER: Received from client: /lib/x86_64-linux-gnu/libm.so.6 cos 2
SERVER: dispatcher_logic: Done. Worker thread terminating.
□

```

Unit test 3:

Incorrect argument of dlopen (wrong location of library or library does not exist)

Unit test 4:

Incorrect function in library (function does not exist in library)

Unit test 5:

sin as a function to be executed.

Unit test 6:

tan to be invoked.

Unit test 7:

abs is checked.

Unit test 8:

File limit is checked.

Server:

```

keshav@keshav-Lenovo-ideapad:~/Documents/cs303/assign1$ ./ipc_demo server
8
Creating AF_LOCAL socket at path ./cs303_sock
An old socket file exists, removing it.
Listening for client connections...

Waiting for incoming connections...
SERVER: Creating a worker thread.
Waiting for incoming connections...
SERVER: dispatcher_logic: starting
SERVER: Received from client: /lib/x86_64-linux-gnu/libm.so.6 abs 2
ERROR in opening dll
SERVER: dispatcher_logic: Done. Worker thread terminating.
□

```

Client:

```
keshav@keshav-Lenovo-ideapad:~/Documents/cs303/assign1$ ./ipc_demo client
8
Creating AF_LOCAL socket at path ./cs303_sock
CLIENT: Connect to server, about to write some stuff...
CLIENT: Received from server:: error
CLIENT: Processing done, about to exit...
```