Laboratory 2

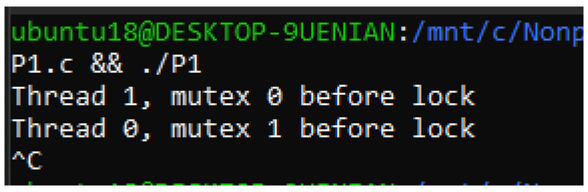
1. Deadlock
    - before practice 1:
        - cat deadlock.c

```
#include <stdio.h>
#include <pthread.h>
#include <unistd.h>

pthread_mutex_t mtx[2];

void * ThrFunc(void * p){
    int * param = (int *) p;
    pthread_mutex_lock(&mtx[*param]);
    pthread_mutex_lock(&mtx[1-*param]);
    return 0;
}

int main(){
    pthread_t thr1;
    pthread_t thr2;
    int i1 = 0, i2 = 1;
    pthread_mutex_init(&mtx[0], NULL);
    pthread_mutex_init(&mtx[1], NULL);
    pthread_create(&thr1, NULL, ThrFunc, &i1);
    pthread_create(&thr2, NULL, ThrFunc, &i2);

    pthread_join(thr1, NULL);
    printf ("first\n");
    pthread_join(thr2, NULL);
    printf ("second\n");
    pthread_mutex_destroy(&mtx[0]);
    pthread_mutex_destroy(&mtx[1]);
    return 0;
}
```

        - gcc -pthread -o deadlock deadlock.c && ./deadlock
        - Run the compiled executable, does it hang? Is the deadlock perfect (meaning does it happen for sure all the time)?
            - It is not for sure that this happened all the time because of this line:
              pthread_mutex_lock(&mtx[1-*param]);
            - Both thread access the code which lock the resources on index 0 and 1 for the first thread and 1 and 0 for the second, so depending on the order in which the threads call the function and get to the locking stage, the process will sometimes hang, sometimes not. If the resources are locked by a thread, freed and then locked by the other thread, the program should work.
        - Practice1:
            -

```
#include <stdio.h>
#include <pthread.h>
#include <unistd.h>

pthread_mutex_t mtx[2];
// count for number of threads
int no;
pthread_mutex_t mutex;
pthread_cond_t condition;

void * ThrFunc(void * p){
    int * param = (int *) p;
    pthread_mutex_lock(&mtx[*param]);
    /*
    pthread_cond_broadcast
```

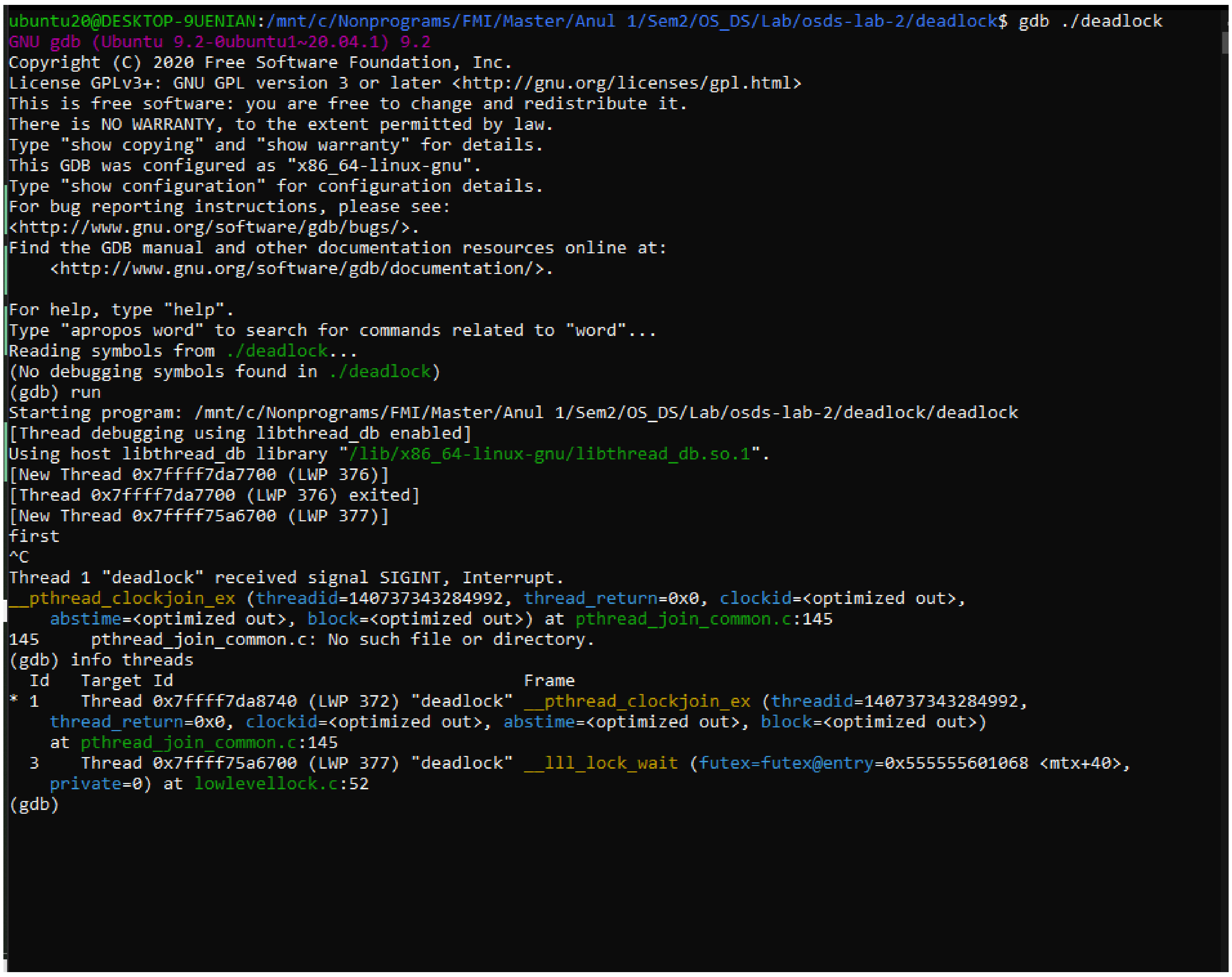
```
            restarts all the threads that are waiting on the condition variable cond.
            Nothing happens if no threads are waiting on cond.
        pthread_cond_wait
            atomically unlocks the mutex (as per pthread_unlock_mutex) and waits
            for the condition variable cond to be signaled. The thread execution is
            suspended and does not consume any CPU time until the condition variable
            is signaled. The mutex must be locked by the calling thread on entrance to
            pthread_cond_wait. Before returning to the calling thread, pthread_cond_wait
            re-acquires mutex (as per pthread_lock_mutex).
        */
        no += 1;
        if (no == 2){
            no = 0;
            // unblocks all threads which are waiting on the condition
            pthread_cond_broadcast(&condition);
        }
        else
            // make all threads wait for condition signal
            while (pthread_cond_wait(&condition, &mutex));

        pthread_mutex_unlock(&mutex);
        printf("Thread %d, mutex %d before lock\n", *param, 1 - *param);
        pthread_mutex_lock(&mtx[1-*param]);
        printf("Thread %d, mutex %d locked\n", *param, 1 - *param);

        return 0;
    }

    int main() {
        pthread_t thr1;
        pthread_t thr2;
        int i1 = 0, i2 = 1;
        pthread_mutex_init(&mutex, NULL);
        pthread_cond_init(&condition, NULL);
        pthread_mutex_init(&mtx[0], NULL);
        pthread_mutex_init(&mtx[1], NULL);
        pthread_create(&thr1, NULL, ThrFunc, &i1);
        pthread_create(&thr2, NULL, ThrFunc, &i2);

        pthread_join(thr1, NULL);
        printf ("first\n");
        pthread_join(thr2, NULL);
        printf ("second\n");
        pthread_mutex_destroy(&mtx[0]);
        pthread_mutex_destroy(&mtx[1]);
        return 0;
    }
```

  - gcc -pthread -o P1 P1.c && ./P1
  - 

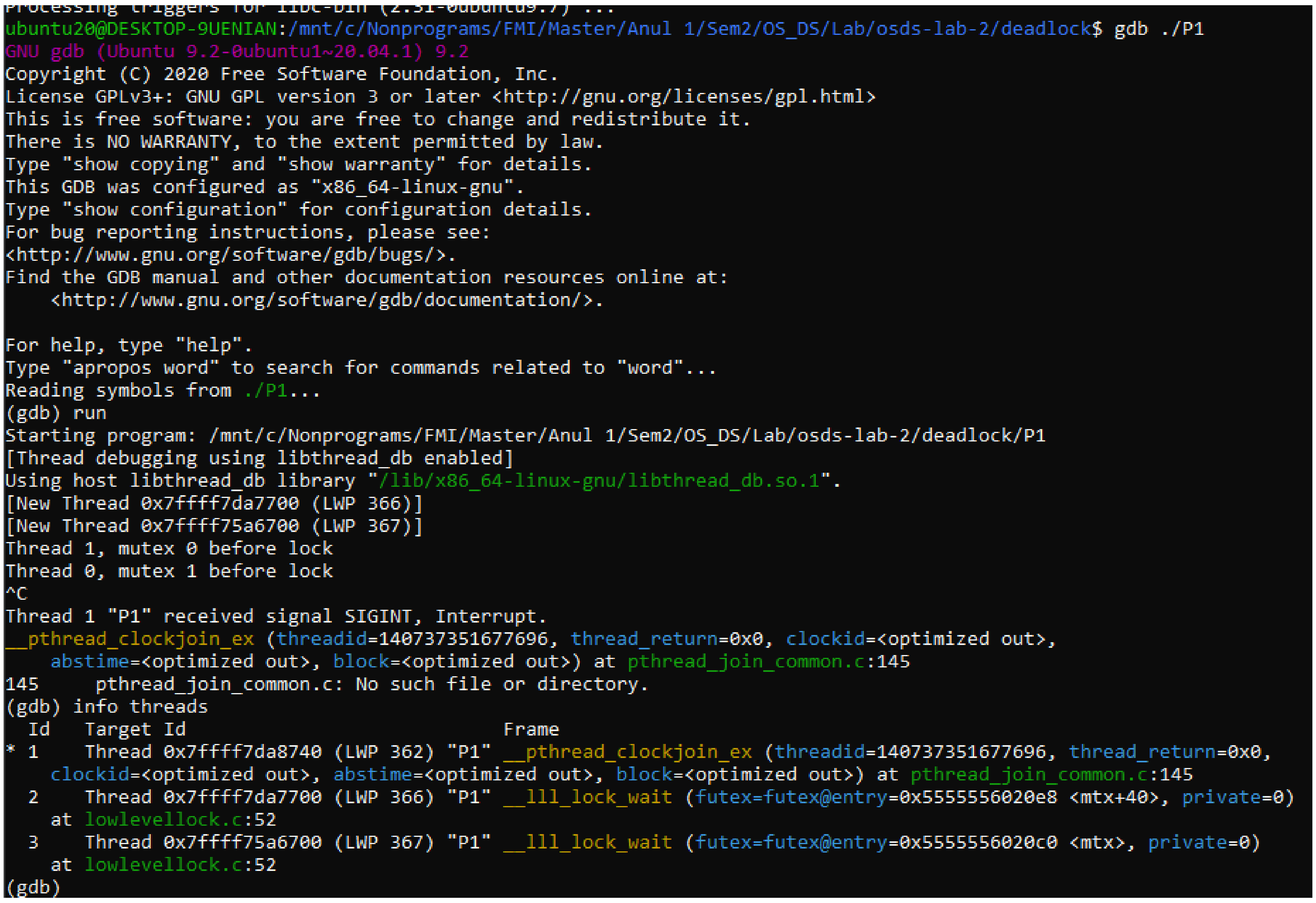
```
ubuntu18@DESKTOP-9UENIAN:/mnt/c/Nonp
P1.c && ./P1
Thread 1, mutex 0 before lock
Thread 0, mutex 1 before lock
^C
```

  -

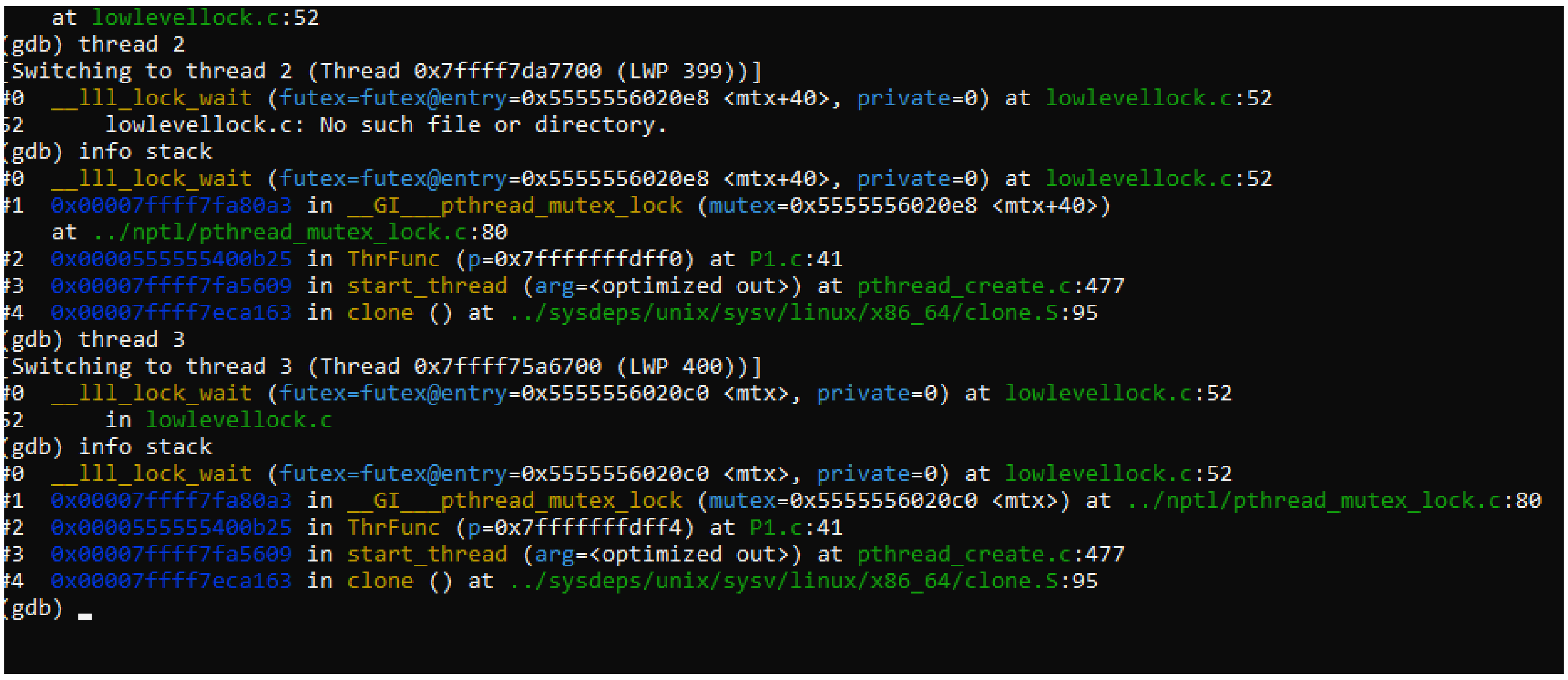
- Practice 2:
  - gcc -pthread -O0 -g -o P1 P1.c
  - gdb ./P1
  - how many threads does it display? Why?
    - There are only 2 threads because only one thread joined main thread

- 

```
ubuntu20@DESKTOP-9UENIAN:/mnt/c/Nonprograms/FMI/Master/Anul 1/Sem2/OS_DS/Lab/osds-lab-2/deadlock$ gdb ./deadlock
GNU gdb (Ubuntu 9.2-0ubuntu1~20.04.1) 9.2
Copyright (C) 2020 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
    <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ./deadlock...
(No debugging symbols found in ./deadlock)
(gdb) run
Starting program: /mnt/c/Nonprograms/FMI/Master/Anul 1/Sem2/OS_DS/Lab/osds-lab-2/deadlock/deadlock
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
[New Thread 0x7ffff7da7700 (LWP 376)]
[Thread 0x7ffff7da7700 (LWP 376) exited]
[New Thread 0x7ffff75a6700 (LWP 377)]
first
^C
Thread 1 "deadlock" received signal SIGINT, Interrupt.
__pthread_clockjoin_ex (threadid=140737343284992, thread_return=0x0, clockid=<optimized out>,
    abstime=<optimized out>, block=<optimized out>) at pthread_join_common.c:145
145     pthread_join_common.c: No such file or directory.
(gdb) info threads
  Id   Target Id                                  Frame
* 1    Thread 0x7ffff7da8740 (LWP 372) "deadlock" __pthread_clockjoin_ex (threadid=140737343284992,
    thread_return=0x0, clockid=<optimized out>, abstime=<optimized out>, block=<optimized out>)
    at pthread_join_common.c:145
  3    Thread 0x7ffff75a6700 (LWP 377) "deadlock" __lll_lock_wait (futex=futex@entry=0x555555601068 <mtx+40>,
    private=0) at lowlevellock.c:52
(gdb)
```

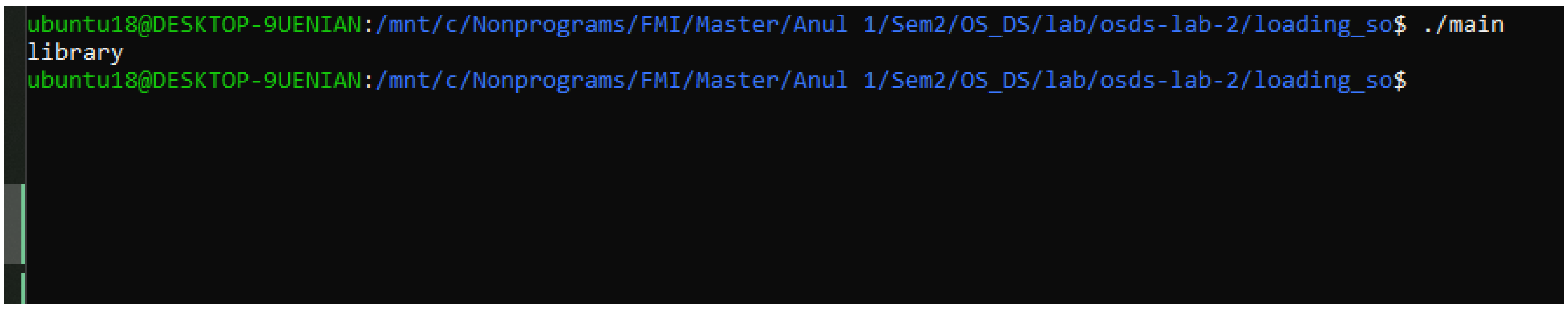
- Do the same steps for the modified deadlock from exercise Practic 1. Does info threads shows you now 3 threads? Why?
  - There 3 threads because none of the threads join the main in the perfect deadlock
  - 

```
Processing triggers for libc-bin (2.31-0ubuntu9.7) ...
ubuntu20@DESKTOP-9UENIAN:/mnt/c/Nonprograms/FMI/Master/Anul 1/Sem2/OS_DS/Lab/osds-lab-2/deadlock$ gdb ./P1
GNU gdb (Ubuntu 9.2-0ubuntu1~20.04.1) 9.2
Copyright (C) 2020 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
    <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ./P1...
(gdb) run
Starting program: /mnt/c/Nonprograms/FMI/Master/Anul 1/Sem2/OS_DS/Lab/osds-lab-2/deadlock/P1
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
[New Thread 0x7ffff7da7700 (LWP 366)]
[New Thread 0x7ffff75a6700 (LWP 367)]
Thread 1, mutex 0 before lock
Thread 0, mutex 1 before lock
^C
Thread 1 "P1" received signal SIGINT, Interrupt.
__pthread_clockjoin_ex (threadid=140737351677696, thread_return=0x0, clockid=<optimized out>,
    abstime=<optimized out>, block=<optimized out>) at pthread_join_common.c:145
145     pthread_join_common.c: No such file or directory.
(gdb) info threads
  Id   Target Id                            Frame
* 1    Thread 0x7ffff7da8740 (LWP 362) "P1" __pthread_clockjoin_ex (threadid=140737351677696, thread_return=0x0,
    clockid=<optimized out>, abstime=<optimized out>, block=<optimized out>) at pthread_join_common.c:145
  2    Thread 0x7ffff7da7700 (LWP 366) "P1" __lll_lock_wait (futex=futex@entry=0x5555556020e8 <mtx+40>, private=0)
    at lowlevellock.c:52
  3    Thread 0x7ffff75a6700 (LWP 367) "P1" __lll_lock_wait (futex=futex@entry=0x5555556020c0 <mtx>, private=0)
    at lowlevellock.c:52
(gdb)
```

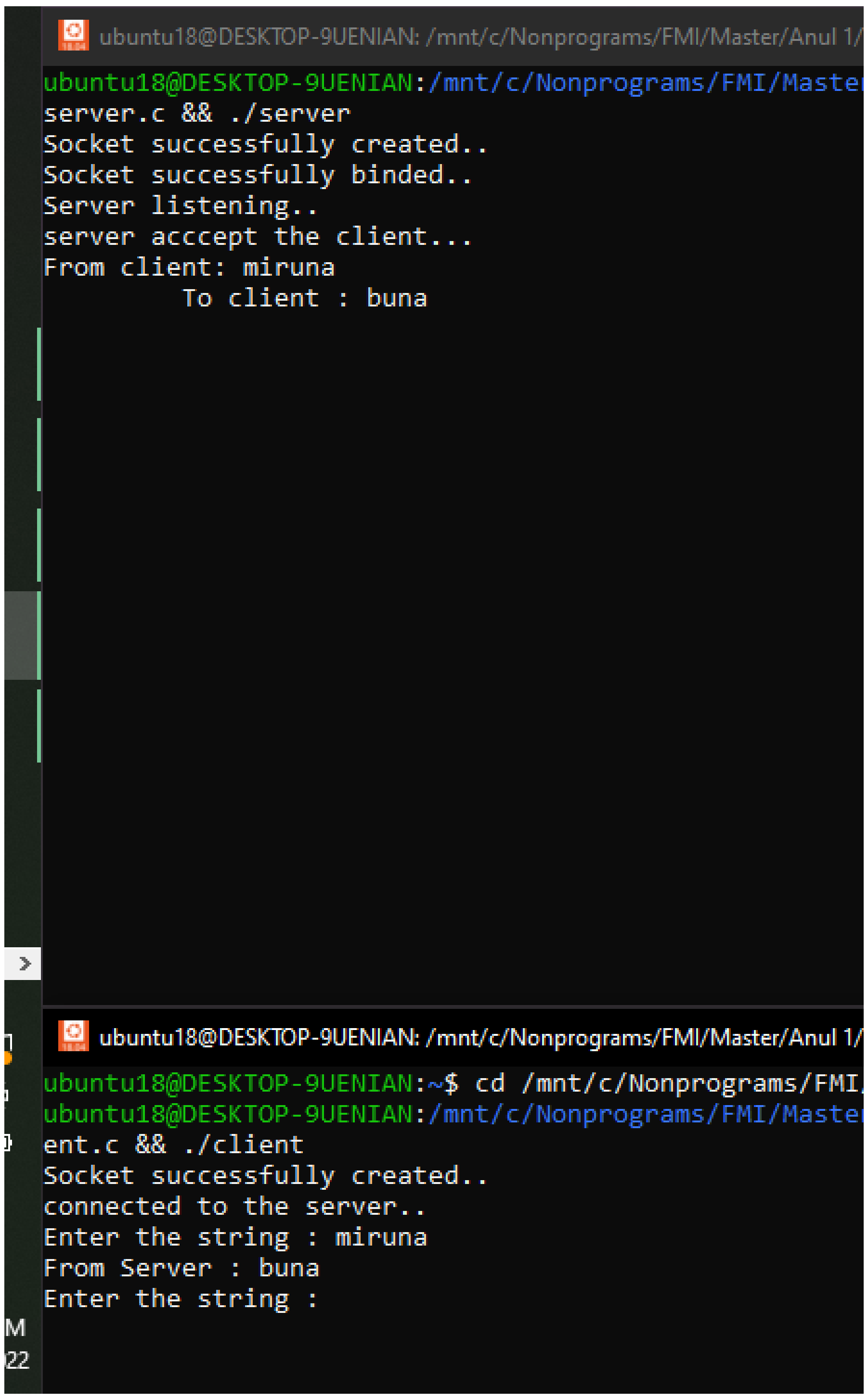
- 

```
    at lowlevellock.c:52
(gdb) thread 2
[Switching to thread 2 (Thread 0x7ffff7da7700 (LWP 399))]
#0  __lll_lock_wait (futex=futex@entry=0x5555556020e8 <mtx+40>, private=0) at lowlevellock.c:52
52      lowlevellock.c: No such file or directory.
(gdb) info stack
#0  __lll_lock_wait (futex=futex@entry=0x5555556020e8 <mtx+40>, private=0) at lowlevellock.c:52
#1  0x00007ffff7fa80a3 in __GI___pthread_mutex_lock (mutex=0x5555556020e8 <mtx+40>)
    at ../nptl/pthread_mutex_lock.c:80
#2  0x0000555555400b25 in ThrFunc (p=0x7fffffffdff0) at P1.c:41
#3  0x00007ffff7fa5609 in start_thread (arg=<optimized out>) at pthread_create.c:477
#4  0x00007ffff7eca163 in clone () at ../sysdeps/unix/sysv/linux/x86_64/clone.S:95
(gdb) thread 3
[Switching to thread 3 (Thread 0x7ffff75a6700 (LWP 400))]
#0  __lll_lock_wait (futex=futex@entry=0x5555556020c0 <mtx>, private=0) at lowlevellock.c:52
52      in lowlevellock.c
(gdb) info stack
#0  __lll_lock_wait (futex=futex@entry=0x5555556020c0 <mtx>, private=0) at lowlevellock.c:52
#1  0x00007ffff7fa80a3 in __GI___pthread_mutex_lock (mutex=0x5555556020c0 <mtx>) at ../nptl/pthread_mutex_lock.c:80
#2  0x0000555555400b25 in ThrFunc (p=0x7fffffffdff4) at P1.c:41
#3  0x00007ffff7fa5609 in start_thread (arg=<optimized out>) at pthread_create.c:477
#4  0x00007ffff7eca163 in clone () at ../sysdeps/unix/sysv/linux/x86_64/clone.S:95
(gdb)
```

-

1. Library hijacking
    - Practice 3:
        - gcc -o library.so -shared -fPIC library.c
        - gcc -o main main.c -ldl && ./main
        - 

```
ubuntu18@DESKTOP-9UENIAN:/mnt/c/Nonprograms/FMI/Master/Anul 1/Sem2/OS_DS/lab/osds-lab-2/loading_so$ ./main
library
ubuntu18@DESKTOP-9UENIAN:/mnt/c/Nonprograms/FMI/Master/Anul 1/Sem2/OS_DS/lab/osds-lab-2/loading_so$
```

        - gcc -o server server.c && ./server
        - gcc -o client client.c && ./client
        - 

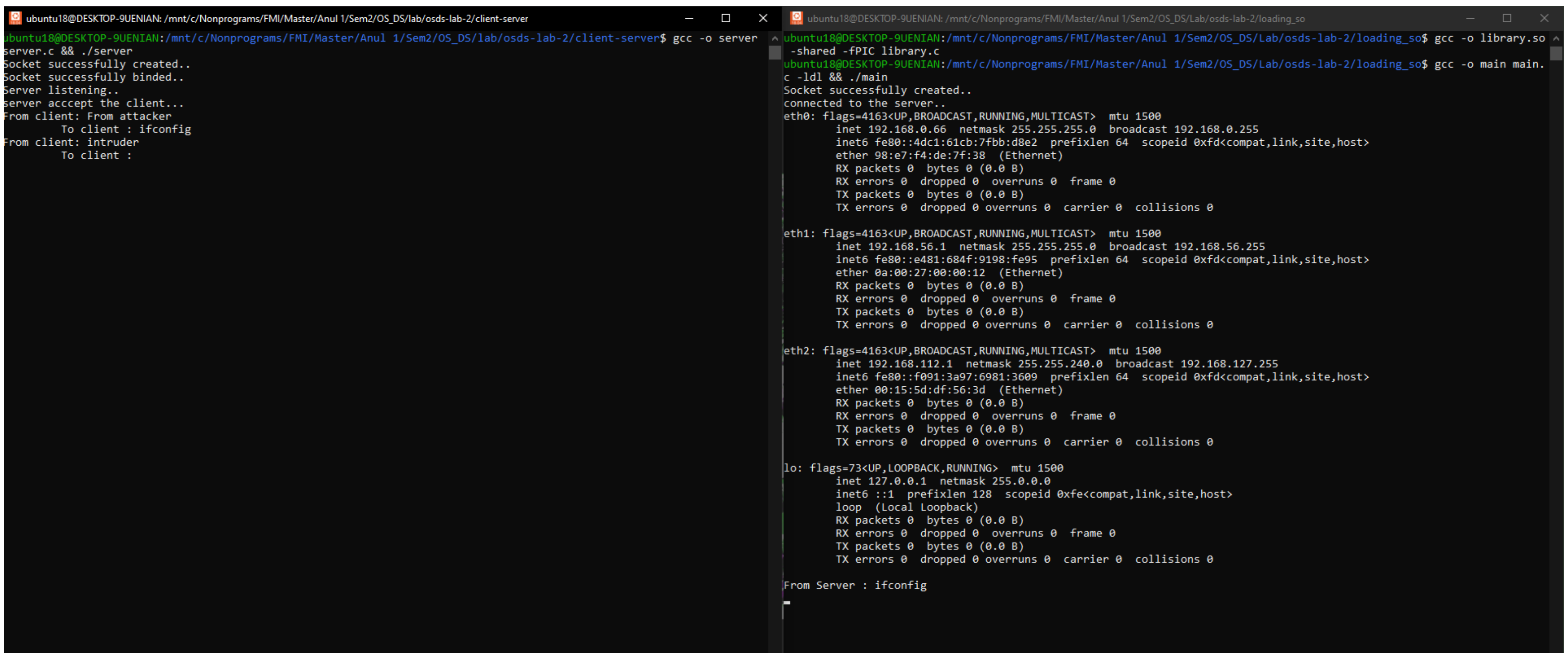
ubuntu18@DESKTOP-9UENIAN: /mnt/c/Nonprograms/FMI/Master/Anul 1/

```
ubuntu18@DESKTOP-9UENIAN:/mnt/c/Nonprograms/FMI/Maste
server.c && ./server
Socket successfully created..
Socket successfully binded..
Server listening..
server acccept the client...
From client: miruna
         To client : buna
```

ubuntu18@DESKTOP-9UENIAN: /mnt/c/Nonprograms/FMI/Master/Anul 1/

```
ubuntu18@DESKTOP-9UENIAN:~$ cd /mnt/c/Nonprograms/FMI
ubuntu18@DESKTOP-9UENIAN:/mnt/c/Nonprograms/FMI/Maste
ent.c && ./client
Socket successfully created..
connected to the server..
Enter the string : miruna
From Server : buna
Enter the string :
```

        - I add the client code in library, recompiled library and recompile + execute main

- 

ubuntu18@DESKTOP-9UENIAN: /mnt/c/Nonprograms/FMI/Master/Anul 1/Sem2/OS_DS/lab/osds-lab-2/client-server
ubuntu18@DESKTOP-9UENIAN: /mnt/c/Nonprograms/FMI/Master/Anul 1/Sem2/OS_DS/Lab/osds-lab-2/loading_so

```
ubuntu18@DESKTOP-9UENIAN:/mnt/c/Nonprograms/FMI/Master/Anul 1/Sem2/OS_DS/lab/osds-lab-2/client-server$ gcc -o server
server.c && ./server
Socket successfully created..
Socket successfully binded..
Server listening..
server acccept the client...
From client: From attacker
         To client : ifconfig
From client: intruder
         To client :
```

```
ubuntu18@DESKTOP-9UENIAN:/mnt/c/Nonprograms/FMI/Master/Anul 1/Sem2/OS_DS/Lab/osds-lab-2/loading_so$ gcc -o library.so
 -shared -fPIC library.c
ubuntu18@DESKTOP-9UENIAN:/mnt/c/Nonprograms/FMI/Master/Anul 1/Sem2/OS_DS/Lab/osds-lab-2/loading_so$ gcc -o main main.
c -ldl && ./main
Socket successfully created..
connected to the server..
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.66  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::4dc1:61cb:7fbb:d8e2  prefixlen 64  scopeid 0xfd<compat,link,site,host>
        ether 98:e7:f4:de:7f:38  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.1  netmask 255.255.255.0  broadcast 192.168.56.255
        inet6 fe80::e481:684f:9198:fe95  prefixlen 64  scopeid 0xfd<compat,link,site,host>
        ether 0a:00:27:00:00:12  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.112.1  netmask 255.255.240.0  broadcast 192.168.127.255
        inet6 fe80::f091:3a97:6981:3609  prefixlen 64  scopeid 0xfd<compat,link,site,host>
        ether 00:15:5d:df:56:3d  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 1500
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0xfe<compat,link,site,host>
        loop  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

From Server : ifconfig
```

-
-